Федеральное государственное автономное
образовательное учреждение
высшего образования
«СИБИРСКИЙ ФЕДЕРАЛЬНЫЙ УНИВЕРСИТЕТ»

Институт космических и информационных технологий

Кафедра систем искусственного интеллекта

УТВЕРЖДАЮ
Заведующий кафедрой
_____ Г. М. Цибульский

« _____» _____ 2017 г.

БАКАЛАВРСКАЯ РАБОТА

09.03.02 «Информационные системы и технологии»

Разработка автоматизированного рабочего места диспетчера учебно–организационного отдела ИКИТ СФУ

Руководитель _____ ст. преподаватель каф. СИИ К. В. Раевич
подпись, дата

Выпускник _____ О. Д. Мухина
подпись, дата

Нормоконтролер _____ М. А. Аникьева
подпись, дата

Красноярск 2017

СОДЕРЖАНИЕ

ВВЕДЕНИЕ

Применение информационных технологий позволяют автоматизировать задачи создания условий для свободного функционирования и развития системы высшего образования. В высших учебных заведениях ежегодно переводятся, восстанавливаются и выходят из академического отпуска большое количество студентов. Временные сроки документооборота определяются ежегодным графиком выполнения работ и представления документов для организации учебного процесса [1].

В ФГАОУ ВО «Сибирский федеральный университет» (далее - СФУ) «Институт космических и информационных технологий» (далее - ИКИТ) существует учебно–организационный отдел (далее - УОО) для организации работы между УОО и обучающимися.

На текущий момент в УОО ИКИТ используется автоматизированная система управления (далее - АСУ) «Web–Деканат», которая автоматизирует функции УОО. Система внедрена в структуру УОО с 2008 года по настоящее время. В данной программе не реализована функция автоматической установки разницы учебных планов и, соответственно, автоматического формирования всех необходимых документов (индивидуальных учебных планов, ведомостей, протоколов). В связи с этим сотрудники увеличивают срок подготовки и выхода приказа, тем самым нарушая установленный порядок.

В остальных институтах СФУ функция автоматической установки разницы учебных планов не автоматизирована. Для создания документов используется пакет Microsoft Office. Существует ряд коммерческих программных продуктов, реализующих задачу автоматизации деканатов (как в составе комплексной системы автоматизации ВУЗа, так и в рамках автономного модуля). К наиболее известным относятся «1С:Университет» (модуль «Управление контингентом»), система «TANDEM UNIVERSITY» (модуль «Контингент студентов»), «Галактика Управление Вузом» (Модуль

«Управление контингентом студентов»), «Деканат» лаборатории математического моделирования и информационных систем (далее – ММИС) (г. Шахты). Эти системы позволяют на основе заполненных справочников и баз данных организовывать запросы, генерирующие необходимые в учебной деятельности документы (справки, приказы). Особенность систем в том, что они не направлены на установление разницы в учебных планах. По проведенной статистической работе, деятельность сотрудника УОО (деканатов) по установлению разницы учебных планов в 60 % высших учебных заведениях не автоматизирована, либо находится на низком уровне автоматизации. Это затрудняет работу учебно–организационных отделов.

Применяемый способ решения проблемы представляет собой спроектированный модуль действующей АСУ «Web–Деканат». Модуль предполагает создание базы данных «Движение контингента» и формирование всех необходимых документов в подмодуле «Перезачет оценок». В нашем случае контингент – это совокупность людей, образующих однородную в каком–либо отношении группу или категорию. Движение контингента подразумевает отслеживание информации о перемещении студентов внутри института (восстановление, отчисление). Данная подсистема предназначается для упрощения работы сотрудников института, занимающихся установлением разницы учебных планов. Особенность подсистемы заключается в том, что пользователь имеет доступ к ней 24 часа в сутки.

Темой выпускной квалификационной работы является «Разработка автоматизированного рабочего места диспетчера учебно–организационного отдела ИКИТ СФУ».

В рамках работы решаются следующие задачи:

- анализ предметной области;

- проектирование подмодуля «Перезачет оценок» в АСУ «Web–Деканат» ИКИТ СФУ;

- разработка подмодуля «Перезачет оценок» в АСУ «Web–Деканат» ИКИТ СФУ.

Глава 1. Анализ предметной области

Автоматизация деятельности деканатов (учебно–организационных отделов) является одной из типовых задач информатизации ВУЗа. Многие ВУЗы самостоятельно занимаются разработкой автоматизированных систем для собственных нужд.

1.1 Программные продукты для автоматизации деканатов

На текущий момент в России предлагается ряд коммерческих программных продуктов, реализующих задачу автоматизации деканатов или учебно–организационных отделов (как в составе комплексной системы автоматизации ВУЗа, так и в рамках автономного модуля). К наиболее известным относятся «1С:Университет» (модуль «Управление контингентом»), система «TANDEM UNIVERSITY» (модуль «Контингент студентов»), «Галактика Управление Вузом» (Модуль «Управление контингентом студентов»), «Деканат» лаборатории ММИС (г. Шахты). Эти системы позволяют на основе заполненных справочников и баз данных организовывать запросы, генерирующие необходимые в учебной деятельности документы (справки, приказы и т. п.).

Система «Деканат» лаборатории ММИС (г. Шахты) [2] имеет клиент–серверную архитектуру и включает в себя базу данных на MS SQL Server, клиентское приложение «Деканат» и вспомогательную программу «UsersManager» для администрирования системы. В системе реализована возможность редактирования уже имеющихся отчетов и разработки новых, с использованием MS Word и Excel. По умолчанию имеются такие типовые отчеты, как справка студента, справка для военкомата, личная карточка студента, список группы, экзаменационная ведомость, зачетная ведомость и др. Отчеты в системе «Деканат» представляют собой документ Word или Excel, дополненный помеченными полями для вставки информации из базы данных.

Информационная система «Деканат» предназначена для ведения личных дел студентов и может работать отдельно или в составе ИС «Электронные ведомости». Программа позволяет осуществлять перевод одного или нескольких студентов одновременно из группы в группу, проводить отчисление и восстановление с хранением атрибутов приказов в истории перемещения студента.

Система безопасности построена на базе ролей, которые могут назначаться пользователю для предоставления доступа к заданным факультетам. Пользовательский интерфейс формируется в соответствии с выданными пользователю правами.

ИС «Деканат» интегрируется с системами «Планы», «Приемная комиссия», «Электронные ведомости» и «Тестирование». Для быстрого создания базы студентов поддерживается импорт списков студентов из текстовых или табличных документов.

В продолжение об универсальных продуктах для деканатов рассмотрим вариант решения на платформе «1С: Предприятие 8.3» [3]. Актуальность программной платформы «1С» с каждым днем увеличивается, большинство коммерческих и государственных организаций пользуются данным продуктом, поэтому не удивительно, что компания выпустила решение специально для высших учебных заведений.

Программный продукт «1С: Университет ред.2.0» представляет собой решение для автоматизации управленческой деятельности в учреждениях высшего профессионального образования.

«1С: Университет ред.2.0» разработан на технологической платформе «1С: Предприятие 8.3». Весь функционал продукта доступен в web–клиенте.

Решение позволяет автоматизировать учет, хранение, обработку и анализ информации об основных процессах высшего учебного заведения: поступление в вуз, обучение, оплата за обучение, выпуск и трудоустройство выпускников, расчет и распределение нагрузки профессорско–преподавательского состава, деятельность учебно–методических отделов и

деканатов, поддержка ФГОС–3 и уровневой системы подготовки (бакалавр, специалист, магистр) на уровне учебных планов и документов государственного образца об окончании вуза, формирование отчетности.

В программном продукте «1С: Университет ред.2.0» реализована подсистема «Управление контингентом», которая включает в себя следующие функции: хранение и обработка сведений о контингенте студентов вуза; контроль движения контингента студентов; ведение зачетных книг студентов; учет успеваемости и посещаемости; возможность массового формирования и вывода на печать ведомостей, списков, отчетов и справок.

Выбирая средства для реализации Электронного деканата, многие учебные заведения обращают свой взгляд на СДО Moodle [4]. И это не случайно. Moodle решает эту задачу. Среди ее достоинств кроссплатформенность, русифицированный дружественный интерфейс, обширная справочная система, широкий набор методов подачи материала. Одним из основных достоинств является универсальность с точки зрения организации учебного процесса – СДО Moodle реализует среду обучения, в которой студенты могут взаимодействовать с учебными материалами, с преподавателями и друг с другом. Это является ключом к универсальности Moodle, позволяя применять эту систему для организации самых разных видов обучения в организациях разных типах.

Основной функцией «Электронного деканата» реализованного на СДО Moodle является хранение и обработка информации о ходе учебного процесса и его участниках, а также автоматизация взаимодействия между тремя участниками учебного процесса по электронной форме обучения: администрация – преподаватель – студент. «Электронный деканат» позволяет выполнять ряд функций традиционного деканата на более технологичном уровне, значительно сокращая время выполнения определенной работы традиционного деканата [5].

Сибирский государственный технологический университет (СибГТУ), ныне опорный университет Красноярского края – Сибирский государственный

аэрокосмический университет имени академика М.Ф. Решетнева (СибГАУ) так же разработал «Электронный деканат» на базе СДО Moodle. Программное средство предназначено для просмотра результатов сессий и пропусков занятий в семестре студентов факультета высшего учебного заведения. Организован удаленный доступ к этим данным посредством Web–приложения. Программное средство внедрено на факультете автоматизации и информационных технологий СибГТУ (далее – СибГАУ) и успешно применяется в работе деканата.

Система «Электронный деканат» (далее – СЭД) СибГТУ объединяет в себя информационно–аналитическую интегрированную систему управления ВУЗом (далее – ИИАС), СДО Moodle, систему модульно–рейтинговой технологии обучения (далее – МРТО) [6].

Следует отметить, что реализация электронных деканатов посредством СДО Moodle широко распространена. Ею пользуются ведущие томские университеты, такие как Национальный исследовательский томский политехнический университет и Томский государственный университет систем управления и радиоэлектроники. Данных об этих системах в открытом доступе нет.

1.2 Обзор автоматизированной информационной системы в СФУ

Сибирский федеральный университет (СФУ) насчитывает в своем составе двадцать институтов. Для автоматизации учебно–организационных отделов (деканатов) была внедрена система АСУ «Деканат». Она включает в себя базу данных всех студентов СФУ, позволяет оформлять и проводить различные приказы и отслеживать успеваемость, посредством учета электронных ведомостей. Так же данная система позволяет просматривать личную информацию о каждом студенте (личную карточку), отслеживать историю перемещений, зачислять и отчислять обучающихся. В данной системе доступны все необходимые виды отчетов, которые можно импортировать в файлы Microsoft Office.

В целях безопасности доступ предоставляется каждому институту индивидуально. Основной недостаток данной программы – локальный доступ и отсутствие возможности автоматизировано устанавливать разницу в учебных планах.

В Институтах Сибирского федерального университета установление разницы в учебных планах регламентируется положением о порядке перевода, восстановления и отчисления студентов Федерального государственного автономного образовательного учреждения высшего профессионального образования «Сибирский федеральный университет», утвержденное на заседании ученого совета — протокол № 1 от 27.01.2014 [7].

Установление разницы в учебных планах в различных институтах СФУ имеет свои различия, не противоречащие действующему положению. По авторитетному мнению сотрудника Учебного управления сибирского федерального университета Пилипенко Ю. А., во всех институтах университета использование информационных систем и технологий в установлении разницы ограничивается использованием пакета Microsoft Office.

1.3 Порядок восстановления, перевода и отчисления в институте космических и информационных технологий

Порядок восстановления и перевода в ИКИТ СФУ определяется положением о порядке перевода, восстановления и отчисления студентов Федерального государственного автономного образовательного учреждения высшего профессионального образования «Сибирский федеральный университет», утвержденным на заседании ученого совета — протокол № 1 от 27.01.2014 [7].

Временные сроки документооборота определяются ежегодным графиком выполнения работ и представления документов для организации учебного процесса [1].

По распоряжению заместителя директора по учебной работе ИКИТ СФУ Ю. Ю. Якунина требуется автоматизировать установление разницы в учебных планах, что не противоречит действующему положению о порядке перевода, восстановления и отчисления студентов.

Учитывая выше сказанное, был разработан следующий порядок установления разницы в учебных планах.

1) Диспетчер УОО ИКИТ СФУ обязан принять заявление на «Сверку учебных планов, в котором будут указаны все необходимые для работы данные: фамилия, имя, отчество, контактный номер телефона, шифр и название направления на котором заявитель ранее обучался, семестр с которого заявитель был отчислен или в котором продолжает обучение (в случае перевода), шифр и название направления на которое заявитель желает восстановиться или перевестись, в случае перевода заявитель обязан указать ВУЗ из которого переводится. Вместе с заявлением заявитель обязан предоставить документ подтверждающий обучение: зачетная книжка или справка об обучении.

2) После получения необходимой информации диспетчер УОО ИКИТ приступает к установлению разницы в учебных планах по следующему алгоритму:

– если названия дисциплин совпадают полностью, а так же совпадают форма контроля и количество часов в диапазоне более 30 %, то данные дисциплины перезасчитываются автоматически;

– если названия дисциплин совпадают полностью, а так же совпадает количество часов в диапазоне более 30 %, но не совпадает форма контроля, то данные дисциплины могут быть перезачтены с согласия преподавателя;

– если названия дисциплин совпадают частично, а так же совпадает форма контроля, но разница в количестве часов более 30 %, то данные дисциплины могут быть перезачтены с согласия преподавателя;

– если названия дисциплин не совпадают, но совпадают формы контроля и совпадает количество часов в диапазоне более 30 %, то данные дисциплины могут быть перезачтены с согласия преподавателя;

– если названия дисциплин не совпадают, не совпадают формы контроля, а так же разница в количестве часов составляет более 30 %, то данные дисциплины включаются в индивидуальный учебный план студента (являются выявленной разницей);

– если необходимые дисциплины отсутствуют в предоставленном заявителем документе, то данные дисциплины включаются в индивидуальный учебный план студента (являются выявленной разницей).

3) После выявления разницы диспетчер УОО ИКИТ СФУ обязан подготовить следующие документы:

– Индивидуальный учебный план (план–график) ликвидации академической разницы;

– Ведомость перезачета дисциплин (с согласием преподавателя – в название документа, данная формулировка не включается);

– Ведомость перезачета дисциплин (дисциплины, которые были перезачтены автоматически и согласованы с заместителем директора по учебной работе – в название документа, данная формулировка не включается).

4) Следующим этапом является подписание документов у заместителя директора по учебной работе, оповещение о готовности и согласование с заявителем подготовленных документов и передача необходимых копий ответственному специалисту за выпуск приказа о восстановлении или переводе.

Важно отметить, что положение о восстановлении, переводе и отчислении студентов действует с 2014 года. До этого момента, не было установленного регламента, касательно документооборота, что отрицательно сказывалось на сроках восстановления и перевода студентов. По рассмотренному положению восстановления и перевода можно выделить три

роли: студент, диспетчер, преподаватель. Все они взаимодействуют друг с другом.

Следует отметить проблемы при выявлении разницы в учебных планах:

– Не автоматизирован процесс установления разницы в учебных планах, для более эффективной работы диспетчера.

– Создание различных ведомостей происходит с помощью пакета Microsoft Office. Это трудоемкий процесс со значительной вероятностью ошибки в сравнении с использованием АСУ для конкретных задач диспетчера;

– Отсутствие автоматизированного назначения преподавателя для согласования перезачета дисциплины.

В ИКИТ уже существует АСУ «Web–Деканат», разработанная на языке программирования Java. Она выполняет множество функций для УОО ИКИТ.

Для того, чтобы решить выявленные проблемы, требуется создать подмодуль «Перезачет оценок» для существующей АСУ «Web–Деканат». Было выявлено, что нецелесообразно создавать отдельное программное обеспечение (ПО) для нужд диспетчера УОО ИКИТ СФУ. Так как, действующая АСУ имеет базу студентов нужную для формирования необходимых документов. Разрабатываемый модуль призван осуществлять работу с БД, где хранится информация о студентах и предприятиях, а также формировать необходимые ведомости.

1.4 Обзор автоматизированной системы управления «Web–Деканат» ИКИТ СФУ

С 2008 года на базе Института космических и информационных технологий СФУ (ИКИТ СФУ) эксплуатируется АИС «Деканат». Система разработана на объектно–ориентированном языке программирования C#. Интерфейс выполнен с помощью WindowsForms - интерфейс программирования приложений, отвечающий за графический интерфейс пользователя и являющийся частью программной платформы Microsoft .NET

Framework. Данная платформа упрощает доступ к элементам интерфейса Microsoft Windows за счет создания обёртки для существующего интерфейса приложений в управляемом коде. Автоматизированная информационная система подверглась значительному изменению, так как была вынесена как web–ресурс с помощью языка программирования Java. Тем самым доступ к АСУ «Web–Деканат» возможен 24 часа в сутки даже вне рабочего места сотрудников УОО ИКИТ СФУ. Осуществление входа в систему отображено на рисунке 1.

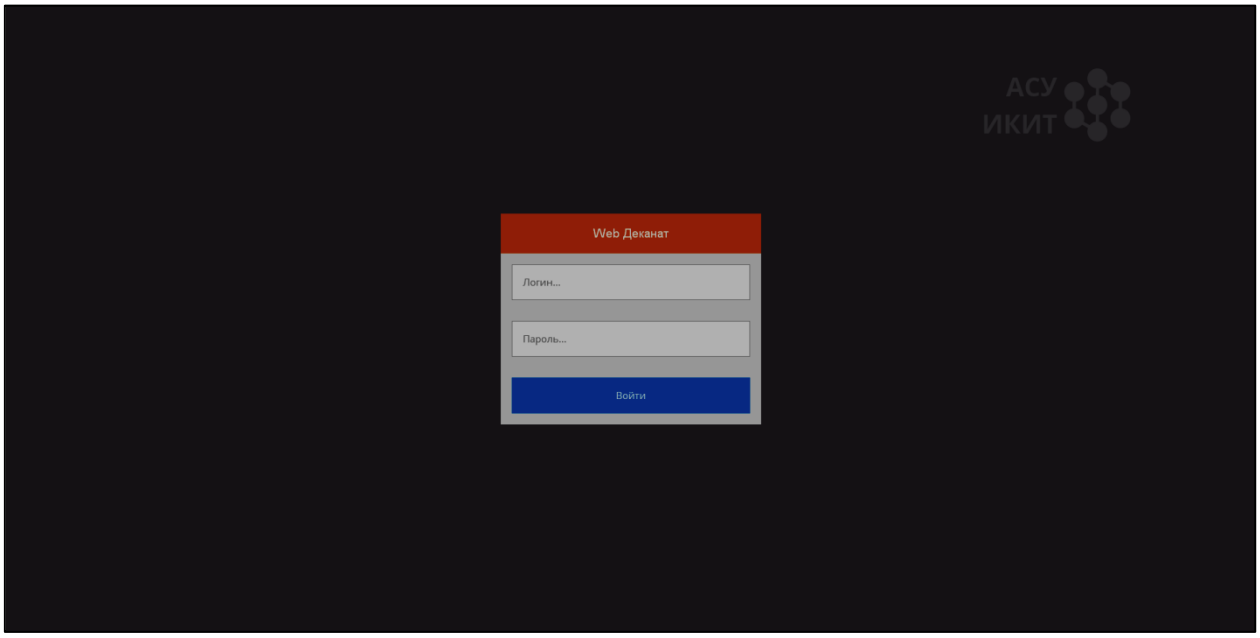


Рисунок 1 — Осуществление входа в систему

После авторизации пользователя открывается основной интерфейс программы. Важно отметить, что в программе доступ к тем или иным возможностям автоматизированной системы управления строго ограничен. Системный администратор создает учетные записи пользователей с определенным уровнем доступа по поручению начальника учебно-организационного отдела института космических и информационных технологий сибирского федерального университета. На рисунке 2 отображены 13 модулей, которые отвечают за различные функции работы диспетчеров УОО ИКИТ СФУ, включая разрабатываемый подмодуль «Перезачет оценок».

Рисунок 2 — Основной интерфейс программы

После изучения функционала данной программы можно выделить основные функции автоматизированной системы управления «Web–Деканат»:

- хранение информации о студентах (номер зачетки, оценки, направление подготовки).
- редактирование диспетчером учебно-организационного отдела инстиута космических и информационных технологий, хранящихся данных;
- формирование различных отчетов (справка директора, список студентов в группе, электронный журнал).

В данной системе присутствует значок модуля «Движение контингента», в которой и будет производиться разработка подмодуля «Перезачет оценок».

Важно отметить, что единственная информация о дисциплине, хранимой в автоматизированной системе управления «Web–Деканат», это оценка за её сдачу или пересдачу студентом – информация отображена на рисунке 3. Просмотр данной информации возможен в электронном паспорте группы, а также в модуле «Комиссии».

В системе реализованы основные функции работы диспетчера, но данные о перезачетах дисциплин не хранятся. Также необходимые документы в АСУ «Web–Деканат» не формируются.

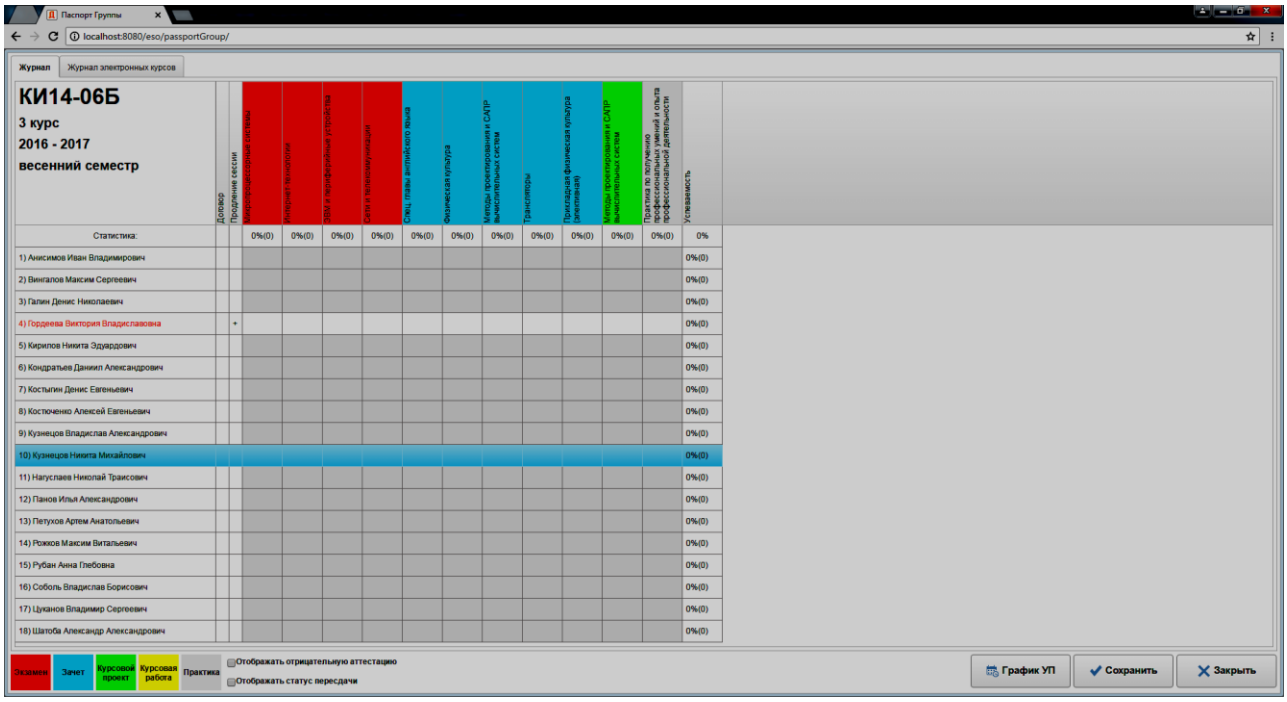


Рисунок 3 — Паспорт группы

При изучении функций данной системы было выявлено, что существует необходимость в автоматизации деятельности диспетчера УОО ИКИТ СФУ ответственного за установление разницы в учебных планах. Это означает, что нужно разработать модуль существующей АСУ «Web–Деканат», для реализации функций, направленных на достижение быстрой и качественной работы по установлению разницы в учебных планах.

1.5 Вывод по главе 1

Широко распространённые комплексные информационные системы или программные решения имеют общий функционал и очень часто нуждаются в индивидуальной доработке. Именно поэтому ВУЗы часто проектируют и разрабатывают подобные информационные системы, используя собственные ресурсы.

Бесспорно, рассмотренные системы в целом отвечают всем современным потребностям высших учебных заведений. Но в каждой рассмотренной системе отсутствует функция (модуль) установления разницы в учебных планах. Данная функция важна в работе любого деканата, так как в течение

всего учебного года студенты восстанавливаются, переводятся и выходят из академических отпусков. Для формирования соответствующих приказов необходимы следующие виды документов:

- Индивидуальный учебный план (или план–график) ликвидации академической разницы;
- Ведомость перезачета дисциплин (с согласием преподавателя);
- Ведомость перезачета дисциплин (автоматически перезачтенные дисциплины согласованные с заместителе директора по учебной работе).

Ни одно из рассмотренных программных решений не предоставляет возможности формировать данные документы автоматизированно. Для того чтобы решить данную проблему необходимо спроектировать модуль в уже существующей АСУ «Web–Деканат» ИКИТ СФУ.

Глава 2. Проектирование подмодуля «Перезачет оценок»

2.1 Деловые преимущества

В настоящее время создание документов для восстановления, перевода и выхода из академического отпуска производится с помощью текстовых редакторов и табличных процессоров. В сравнении с существующим порядком, новое решение позволит обеспечить режим доступа диспетчеров и повысить быстродействие, обеспечить надёжное хранение данных и более полный охват функций, подлежащих автоматизации. Проблема отсутствия подмодуля «Перезачет оценок» представлена на рисунке 4.

Проблема	отсутствие подмодуля «Перезачет оценок»
затрагивает	диспетчера
Ее следствием является	увеличение времени выпуска приказа на перевод, восстановление или выход из академического отпуска, нарушение нормативного периода
Успешное решение	сформировать подмодуль «Перезачет оценок», где будет реализован поиск студентов, восстановление и перезачет оценок.

Рисунок 4 – Перезачет оценок

Проблема отсутствия таблицы «Перезачет оценок» представлена на рисунке 5.

Для того чтобы решить данную проблему требуется сформировать таблицу «Перезачет оценок», которая будет разбита на две таблицы, они не требуют специальных названий, поэтому условно обозначены «Таблица А» и «Таблица В». Поля «Таблицы А» включают следующие столбцы: дисциплины, семестр, группа, количество часов, оценка. Поля «Таблицы В» включают следующие столбцы: дисциплины, семестр, количество часов, форма контроля, ИУП, перезачет.

Проблема	отсутствие таблицы «Перезачет оценок»
затрагивает	диспетчера
Ее следствием является	сложная и долгая работа создания ведомостей в текстовых редакторах
Успешное решение	сформировать таблицу «Перезачет оценок», которая будет разбита на две, они не требуют специальных названий, поэтому условно обозначим их «Таблица А» и «Таблица В». Поля «Таблицы А» включают: дисциплины, семестр, группа, количество часов, оценка. Поля «Таблицы В»: дисциплины, семестр, количество часов, форма контроля, ИУП, перезачет.

Рисунок 6 – Перезачеты

Проблема неавтоматизированного создания ведомостей и планов представлена на рисунке 6.

Проблема	неавтоматизированное создание ведомостей и планов для формирования приказов
затрагивает	диспетчера
Ее следствием является	значительные временные затраты диспетчером на подготовку документов
Успешное решение	создать шаблоны документов (ведомости, планы), куда автоматически вносится определенная информация с БД. Сформированный отчет появляется во всплывающем окне, откуда его можно сохранить в формате .pdf или распечатать.

Рисунок 5 – Ведомости и планы

В текущем пункте были выделены проблемы и способы их решений, что поможет в дальнейшем проектировании подмодуля «Перезачет оценок» автоматизированной систему управления «Web–Деканат».

2.2 Определение позиций изделия

На рисунке 7 отображены позиции разрабатываемого изделия, то есть описывается определенное различие от существующего порядка.

Для	УОО ИКИТ
Которой	оптимизировать процесс создания приказов для восстановления, перевода и выхода из академического отпуска
(Название продукта)	«Перезачет оценок»
Который	основан на положении о порядке перевода, восстановления и отчисления студентов Федерального государственного автономного образовательного учреждения высшего профессионального образования «Сибирский федеральный университет», утвержденном на заседании ученого совета – протокол №1 от 27.01.2014
В отличие от	существующего порядка создания индивидуальных ведомостей и планов в MS Word, Excel
наш продукт	создает ведомости и планы на основе готовых шаблонов и исключает ошибки при их создании

Рисунок 7 – Позиции изделия

На данном рисунке указано наименование и цель создания программного продукта.

2.3 Описание пользователей

В системе существует один тип пользователя: диспетчер. Права доступа пользователя описаны ниже.

Пользовательская среда: в ИКИТ существует один учебно–организационный отдел. В УОО существует один диспетчер отвечающий за установление разницы в учебных планах и подготовку соответствующих документов, для которого создается АСУ. Время создания одной ведомости/плана — две минуты, для сложных документов — не более пяти

минут. Система будет работать на платформе IBM PC. Операционная система: Microsoft Windows XP и выше. База данных реляционного типа. Профиль диспетчера УОО ИКИТ СФУ представлен на рисунке 8.

Типичный представитель	**Диспетчер**
Тип	пользователь
Описание	пользователь системы, наделенный следующими правами: – сравнение и просмотр данных в таблицах «Учебный план «А»» и «Учебный план «В»»; – создание и просмотр сводной таблицы «Разница учебных планов»; – создание и просмотр таблицы «Перезачеты»; – формирование индивидуальных планов и ведомостей, ведомостей перезачета; – экспорт в форматы .doc и/или Adobe .pdf; – печать документов.
Ответственности	пользователь редактирует данные о студентах; создает и проверяет индивидуальные планы и ведомости, ведомости перезачетов
Критерии успеха	получение правильно оформленных индивидуальных ведомостей и планов, ведомостей перезачетов для создания приказов

Рисунок 8 – Профиль диспетчера

Диспетчер тратит около часа на создание самых простых документов (индивидуальный план и ведомость). УОО нуждается в системе, которая ускорит и оптимизирует процесс.

2.4 Требования к документации

В системе представлены «Руководства пользователя». Они содержат расшифровку всех используемых терминов, описания основных вариантов использования, включая альтернативные сценарии, а также подробный обзор интерфейса программы. Интерактивная справка необходима для разрешения возникших во время работы вопросов. В справке должна быть реализована

возможность поиска информации по ключевым словам, а также вариант представления информации по отдельным позициям меню программы. Справка должна содержать полную и подробную информацию по работе системы.

2.5 Обзор прецедентов

Для разрабатываемой системы предусмотрена одна роль. Выявленные прецеденты представлены на рисунке 9.

Код	**Основной актор**	**Наименование**	**Формулировка**
Д1	Диспетчер	Формирование БД	Этот вариант использования позволяет диспетчеру, внести, удалить, изменить данные о студентах в БД
Д2	Диспетчер	Поиск информации	Могут осуществить поиск студента и/или учебного плана, для получения нужной информации
Д3	Диспетчер	Формирование документов	Могут экспортировать в MS Word, Adobe Reader и распечатать индивидуальные планы, ведомости и ведомости перезачетов

Рисунок 9 – Список вариантов использования

Для более точного выявления требований к разрабатываемой программе необходимо более детально рассмотреть каждый прецедент.

2.5.1 Описание вариантов использования

Краткое описание прецедента «Формирование БД» отображено на рисунке 10.

Д1	Диспетчер	Формирование БД	Этот вариант использования позволяет диспетчеру, внести, удалить, изменить данные о студентах в БД

Рисунок 10 – Прецедент «Формирование БД»

Основное действующее лицо: Диспетчер.

Другие участники прецедента: отсутствуют.

Связи с другими вариантами использования: расширяется прецедентами: «Поиск информации», «Формирование документов», так как по БД осуществляется поиск данных, а документы формируются именно из нее.

Краткое описание: данный вариант использования позволяет диспетчеру внести, удалить, изменить некоторую информацию о студенте в БД.

Описание прецедента «Поиск информации» отображено на рисунке 11.

Д2	Диспетчер	Поиск информации	Могут осуществить поиск студента и/или учебного плана, для получения нужной информации

Рисунок 11 – Прецедент «Поиск информации»

Основное действующее лицо: Диспетчер.

Другие участники прецедента: отсутствуют.

Связи с другими вариантами использования: поиск происходит в БД, поэтому связан с прецедентом «Формирование БД».

Краткое описание: данный вариант использования позволяет диспетчеру осуществить поиск учебного плана и/или студента в таблицах БД, для получения нужной информации.

Описание прецедента «Формирование документов» отображено на рисунке 12.

Д3	Диспетчер	Формирование документов	Могут экспортировать в MS Word, Adobe Reader и распечатать индивидуальные планы, ведомости и ведомости перезачетов

Рисунок 12 – Прецедент «Формирование документов»

Основное действующее лицо: Диспетчер.

Другие участники прецедента: отсутствуют.

Связи с другими вариантами использования: вся информация для создания документов хранится в прецеденте «Формирование БД».

Краткое описание: данный вариант использования позволяет создать и распечатать какой–либо вариант ведомости или/и плана. Этот процесс происходит в пределах 2–5 мин., что достигается за счет наличия готовых шаблонов.

2.5.2 Анализ вариантов использования

Анализ сформулированных вариантов использования показал, что с точки зрения потенциальных рисков и архитектурной значимости наиболее существенными являются прецеденты, связанные с работой диспетчера.

Для дальнейшей детализации будут выбраны два прецедента:

1) Формирование БД;

2) Формирование документов.

2.6 Моделирование разрабатываемой программы

UML — язык графического описания для объектного моделирования в области разработки программного обеспечения, моделирования бизнес–процессов, системного проектирования и отображения организационных структур [8].

UML является языком широкого профиля, это — открытый стандарт, использующий графические обозначения для создания абстрактной модели

системы, называемой UML–моделью. Язык UML был создан для определения, визуализации, проектирования и документирования, в основном, программных систем. UML не является языком программирования, но на основании UML–моделей возможна генерация кода [8].

2.6.1 Структуризация вариантов использования

Диаграмма прецедентов (Use case) — UML–диаграмма, отражающая отношения между акторами и прецедентами и являющаяся составной частью модели прецедентов, позволяющей описать систему на концептуальном уровне [9]. Вариант использования — «Формирование БД» подразумевает внесение, удаление и изменение информации в БД. На рисунке 13 приведена Use Case–диаграмма для данного варианта использования.

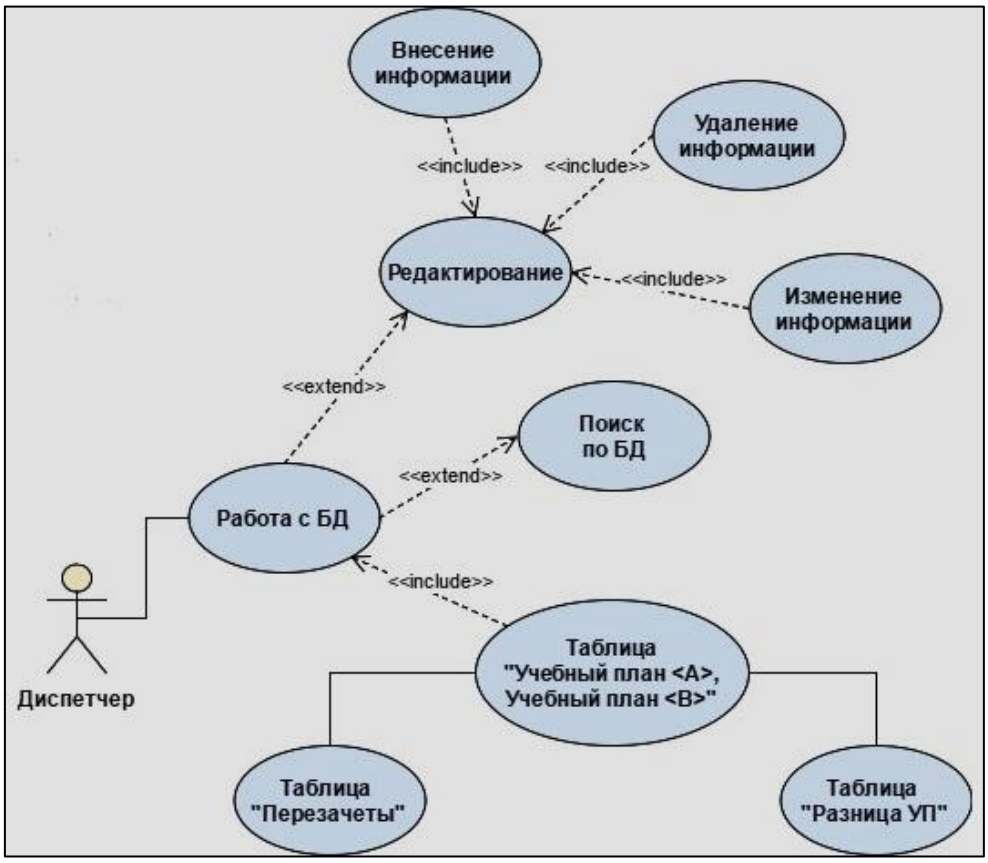


Рисунок 13 — Вариант использования «Формирование БД»

Вариант использования «Формирование документов» основан на прецеденте «Формирование БД», в нем содержится вся информация для

создания ИУП и других документов. На рисунке 14 приведена Use Case–диаграмма для данного варианта использования.

Рисунок 14 — Вариант использования «Формирование документов»

На диаграмме показано, какие действия можно выполнить с индивидуальным учебным планом (далее – ИУП). В данной диаграмме не указано, откуда исходят данные для формирования ИУП, но, очевидно, что информация черпается из прецедента «Формирование БД» Это и полная картина функций показана на рисунке 15.

На данном рисунке показаны актор и сущности разрабатываемого подмодуля «Перезачет оценок» АСУ «Web–Деканат». Рисунок 15 представляет собой совмещение вышеописанных вариантов использования, а именно «Формирование БД», «Формирование документов».

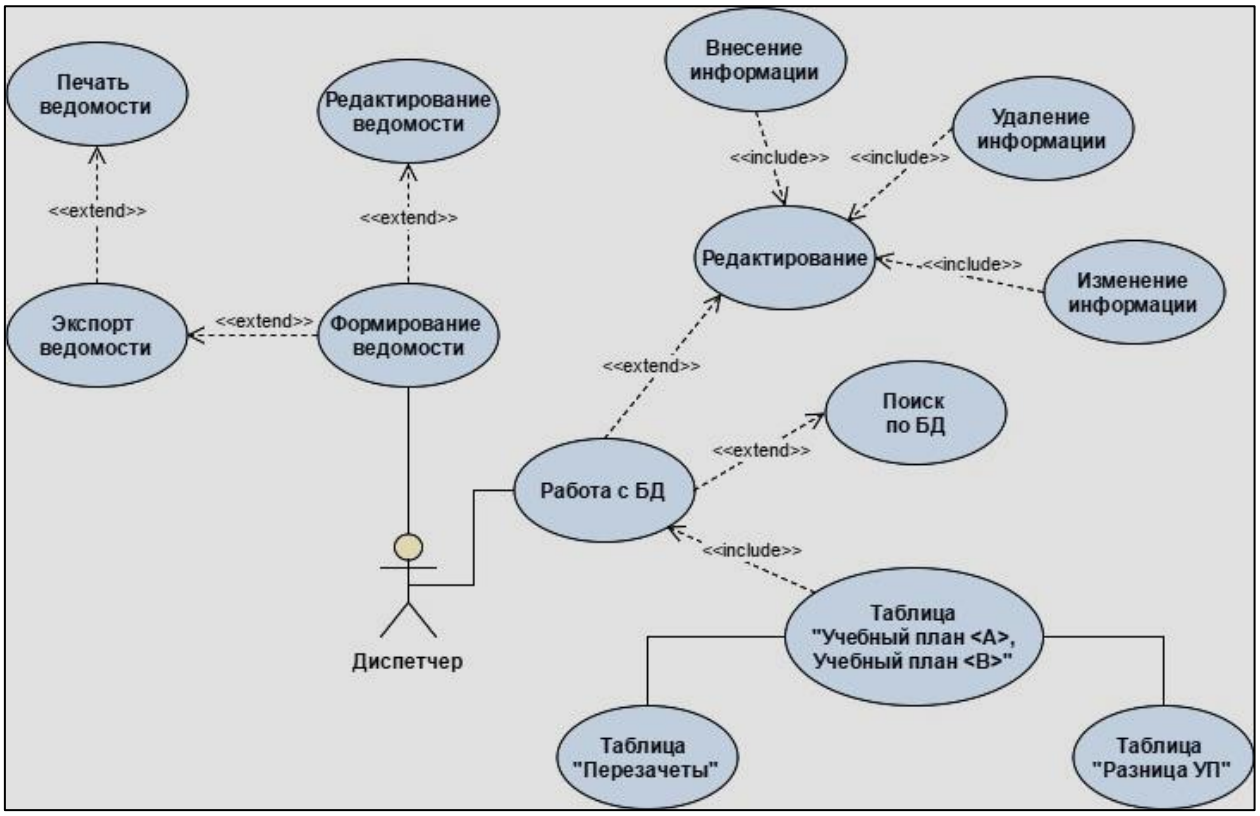


Рисунок 15 — Общая диаграмма вариантов использования

2.6.2 Диаграмма деятельности

Существует два основных прецедента: «Формирование БД», «Формирование документов». Требуется построить диаграммы деятельности каждого из них.

UML (англ. Unified Modeling Language — унифицированный язык моделирования) — язык графического описания для объектного моделирования в области разработки программного обеспечения, моделирования бизнес–процессов, системного проектирования и отображения организационных структур [8].

Диаграмма деятельности — UML–диаграмма, на которой показано разложение некоторой деятельности на её составные части. Под деятельностью (англ. activity) понимается спецификация исполняемого поведения в виде координированного последовательного и параллельного выполнения подчинённых элементов — вложенных видов деятельности и

отдельных действий, соединённых между собой потоками, которые идут от выходов одного узла ко входам другого [10].

Диаграммы деятельности используются при моделировании бизнес–процессов, технологических процессов, последовательных и параллельных вычислений. В данном случае деятельность — документооборот по формированию разницы в учебных планах, составные части — диспетчер и АИС. На рисунке 16 показана диаграмма деятельности прецедента «Формирование БД».

Прецедент начинается, когда диспетчер получает заявление написанное студентом.

Базовый поток — произвести установление разницы в учебных планах:

1) Пользователь открывает подмодуль «Перезачет оценок» в АСУ «Web–Деканат».

2) Система отображает интерфейс программы.

3) Диспетчер находит нужного студента и учебные планы, с помощью поиска «ФИО» или нужной группы.

4) Система отображает интерфейс программы, в котором диспетчер может внести изменения.

5) Система сохраняет изменения, что является окончанием прецедента.

Время сохранения изменений в БД не должно превышать 1 минуты.

Перед тем как начинается этот прецедент, диспетчер должен авторизироваться в автоматизированной системе управления «Web–Деканат», чтобы получить доступ к работе с базой данных.

После окончания прецедента диспетчер завершает работу с базой данных и может сформировать индивидуальный учебный план ликвидации академической разницы, ведомость перезачета дисциплин для согласования с преподавателями или ведомость автоматического перезачета дисциплин, что описано в следующем прецеденте.

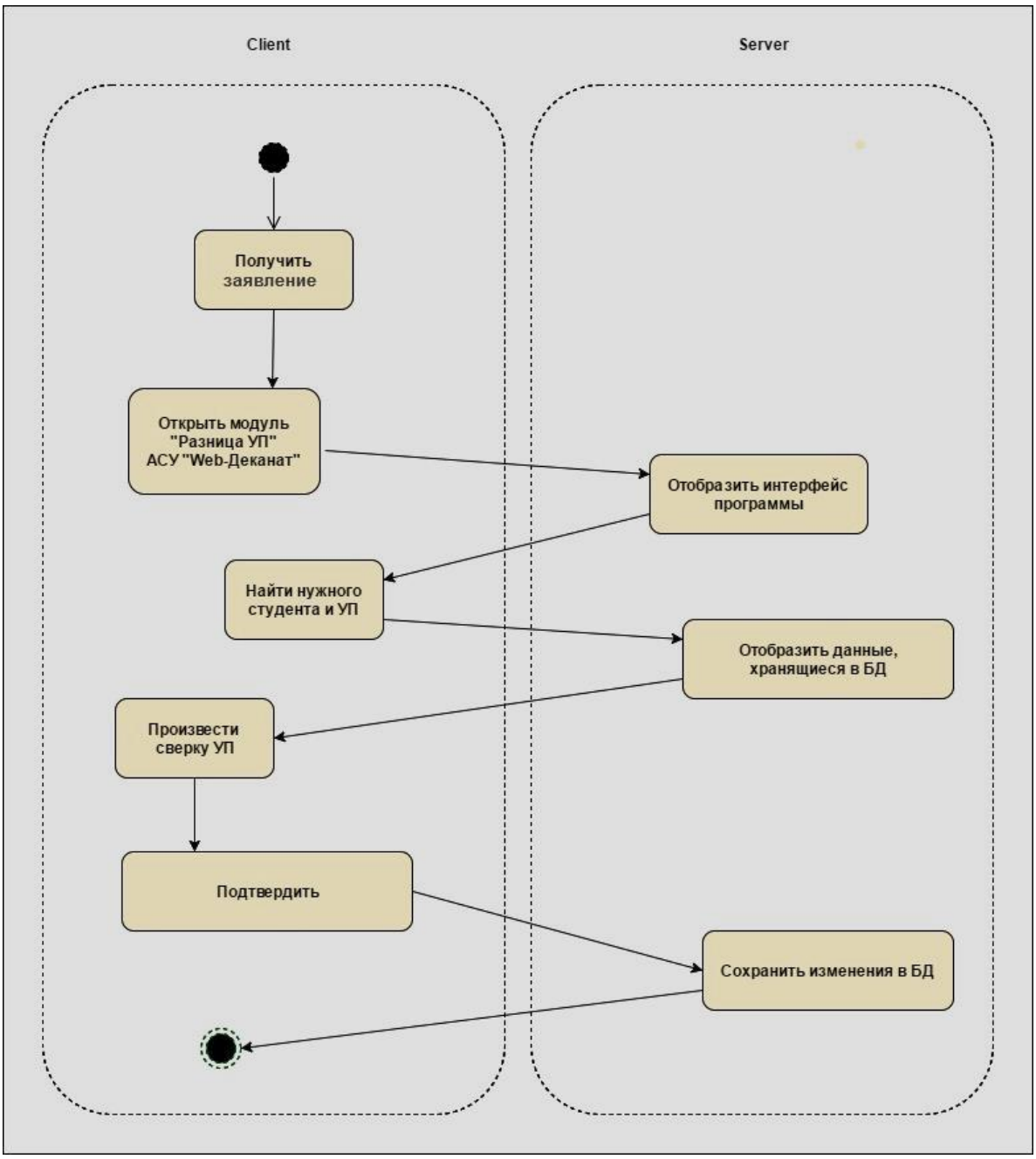


Рисунок 16 — Диаграмма деятельности прецедента «Формирование БД»

Диаграмма деятельности прецедента «Формирование документов» показана на рисунке 17.

Прецедент начинается после формирования БД.

Базовый поток — сформировать ИУП/ведомость.

1) Пользователь открывает подмодуль «Перезачет оценок» АСУ «Web–Деканат».

2) Система отображает интерфейс программы.

3) Пользователь нажимает кнопку «Перезачет оценок», система отображает ее интерфейс и предоставляет выбор атрибутов для формирования разницы учебных планов, затем предоставляет возможность создать необходимые документы.

4) Пользователь осуществляет выбор, система формирует документ и выводит результат на экран.

5) Пользователь экспортирует в файл сформированный документ и может распечатать его.

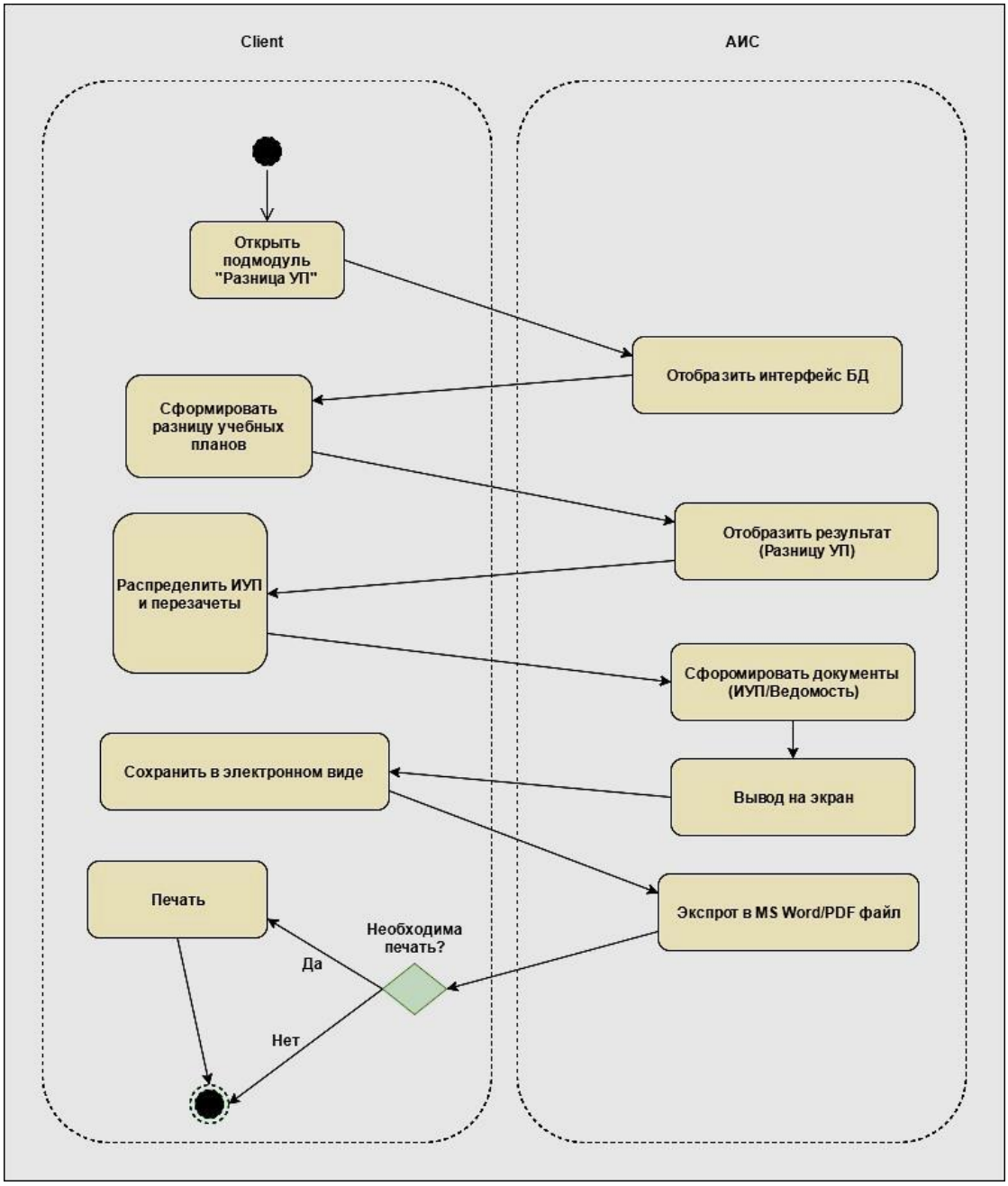


Рисунок 17 — Диаграмма деятельности прецедента «Формирование документов»

Время формирования документа 2–5 минут. Перед тем как начинается этот прецедент, пользователь должен авторизироваться в АСУ «Web–Деканат», чтобы получить доступ к работе с документами. После окончания прецедента пользователь завершает работу с документами. Далее пользователь может повторить прецеденты, либо выйти из системы. Важно отметить, что без выбора студента/группы нельзя сформировать ИУП и ведомости.

2.6.3 Диаграмма классов

Диаграмма классов — диаграмма, демонстрирующая классы системы, их атрибуты, методы и взаимосвязи между ними [11]. Диаграмма классов разрабатываемой системы представлена на рисунке 18.

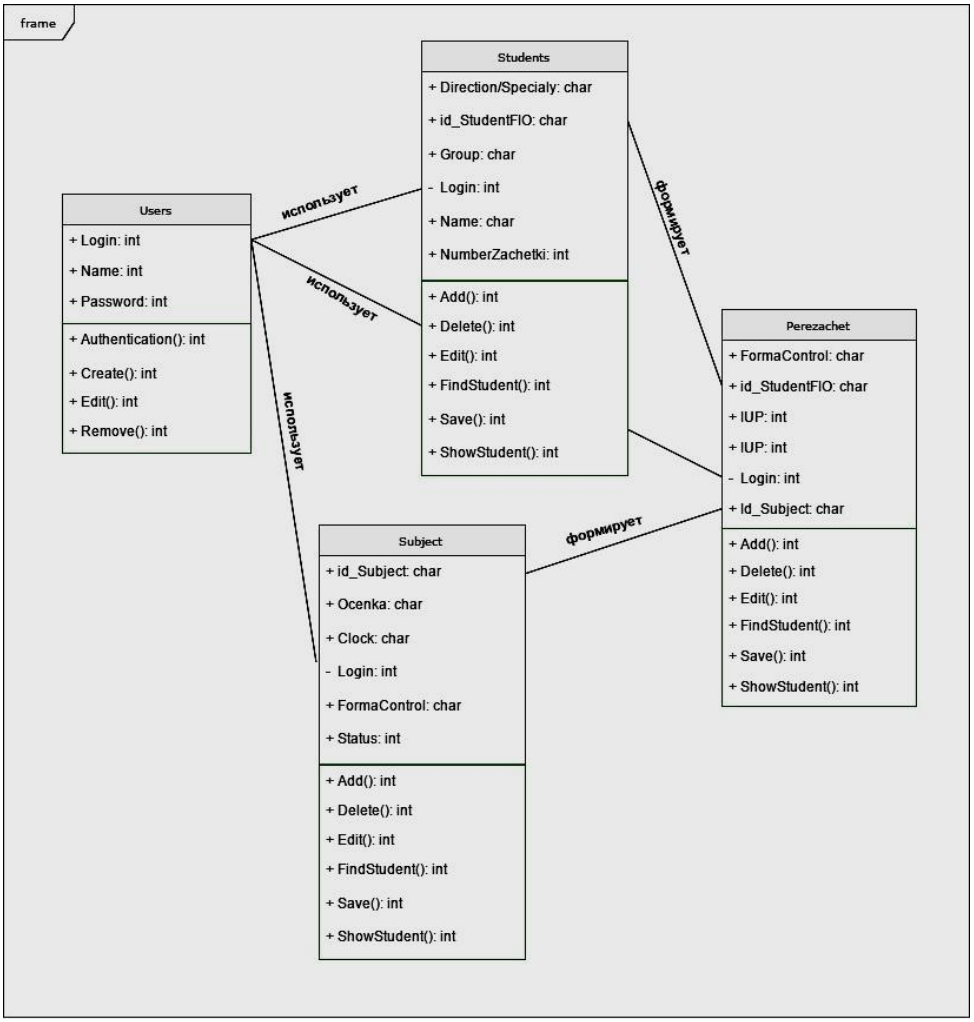


Рисунок 18 — Диаграмма классов подмодуля «Перезачет оценок»

На представленной диаграмме находится четыре класса. Класс «Users» использует остальные классы по ключевому полю Login. Основная функция этого класса авторизировать пользователя и обеспечить доступ к БД и формированию отчетов, а также защитить от несанкционированного доступа к системе. Авторизация происходить с помощью операции «Authentication» и атрибутов Login, Password.

Класс «Perezachet» является сводным для классов «Subject» «Students», но также имеет атрибуты о сроках и типе долга. Этот класс нужен для формирования документов. Здесь устанавливается связь между таблицами «Subject» и «Students». В классе «Perezachet» можно внести, изменить и удалить информацию с помощью операций указанных на диаграмме.

Классы «Subject» и «Students» также имеют операции и атрибуты. Важно отметить, что любое внесение изменений сохраняется программой. А также есть функция по осуществления поиска данных.

2.6.4 Диаграмма последовательностей

Диаграмма последовательности — диаграмма, на которой показано взаимодействие объектов (обмен между ними сигналами и сообщениями), упорядоченное по времени, с отражением продолжительности обработки и последовательности их проявления [12].

На рисунке 19 представлена последовательность выполняемых действий прецедента «Формирование БД». Manager — это диспетчер, BigData — база данных. Сплошная стрелка показывает, какое действие совершает диспетчер, а пунктирная возвращает ответ системы.

Поток событий:

1) Диспетчер запрашивает предприятия.

2) Система показывает предприятия с помощью интерфейса.

3) Диспетчер изменяет данные в БД, а система сохраняет их.

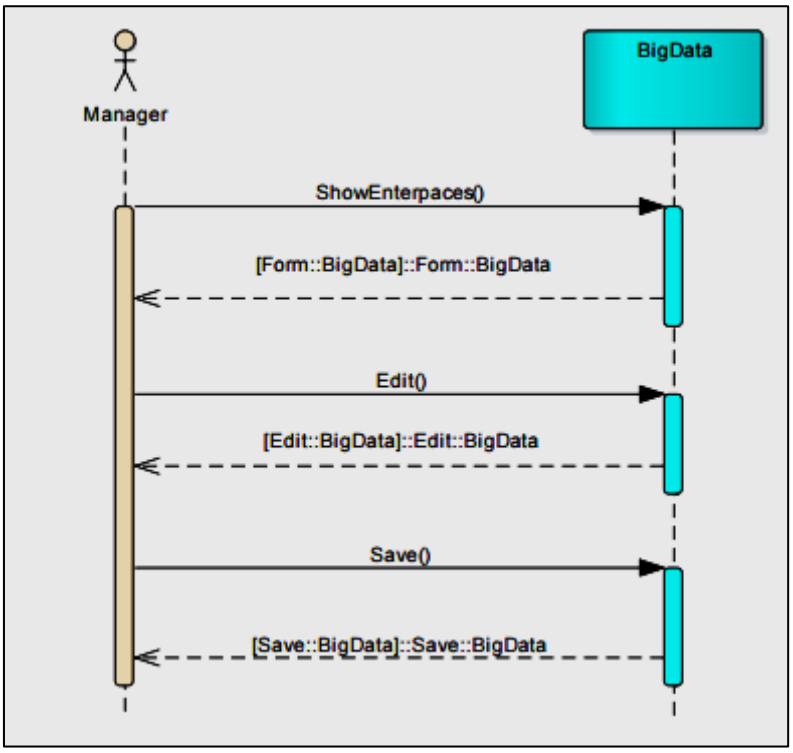


Рисунок 19 — Диаграмма последовательности прецедента «Формирование БД»

Данная последовательность может повторяться по производственной необходимости диспетчера. Важно отметить, что в данной диаграмме вместо «Edit» могут быть операции «Add», «Delete», в зависимости от задачи диспетчера.

На рисунке 20 представлена диаграмма последовательности прецедента «Формирование документов». В данной диаграмме Manager является диспетчером, а Perezachet — это класс нужный для создания документов.

Поток событий:

1) Пользователь запрашивает интерфейс для формирования документа.

2) Система показывает инструменты для создания документа с помощью интерфейса (Form1).

3) Пользователь делает запрос сформировать документ, система его создает и показывает.

4) Пользователь делает запрос на экспорт документа в файл.

5) Система экспортирует документ в формат .docx или .pdf.

6) Пользователь делает запрос печати документа, система организует успешный результат данного действия.

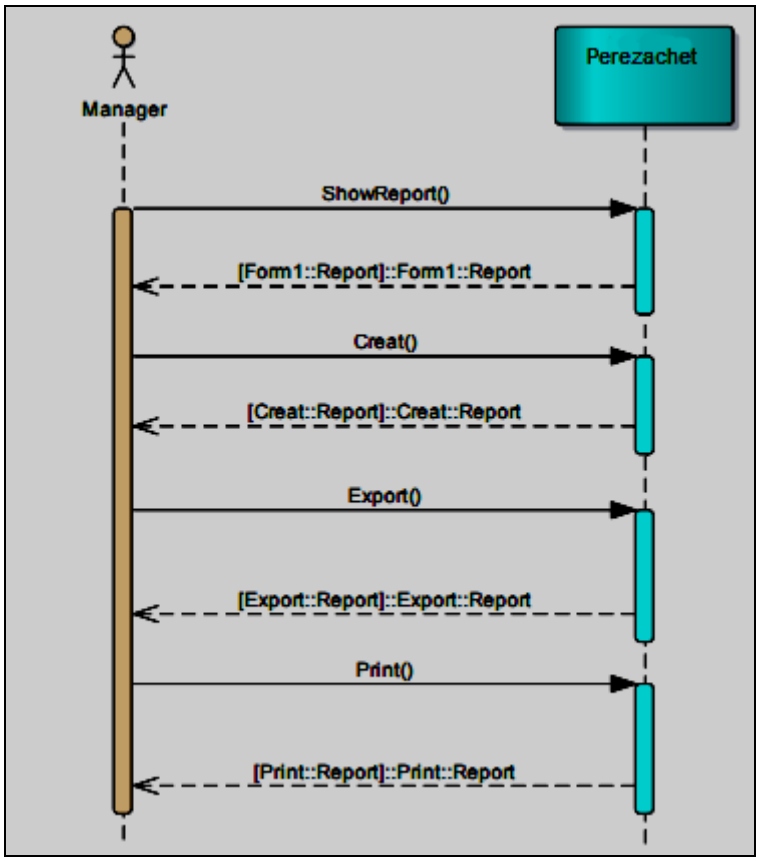


Рисунок 20 — Диаграмма последовательности прецедента «Формирование документов»

Печать является необязательной частью диаграммы последовательности. Данная функция остается на усмотрение пользователя.

2.6.5 Диаграмма состояний

Диаграмма состояний является графом специального вида, который представляет некоторый автомат. Вершинами графа являются возможные состояния автомата, изображаемые соответствующими графическими символами, а дуги обозначают его переходы из состояния в состояние [13].

На практике вершины обычно изображаются в виде окружностей и, если нужно, двойных окружностей. В нотации UML (где UML - язык графического описания для объектного моделирования в области разработки программного обеспечения, моделирования бизнес-процессов, системного проектирования и

отображения организационных структур) состояния изображаются прямоугольниками с закругленными углами.

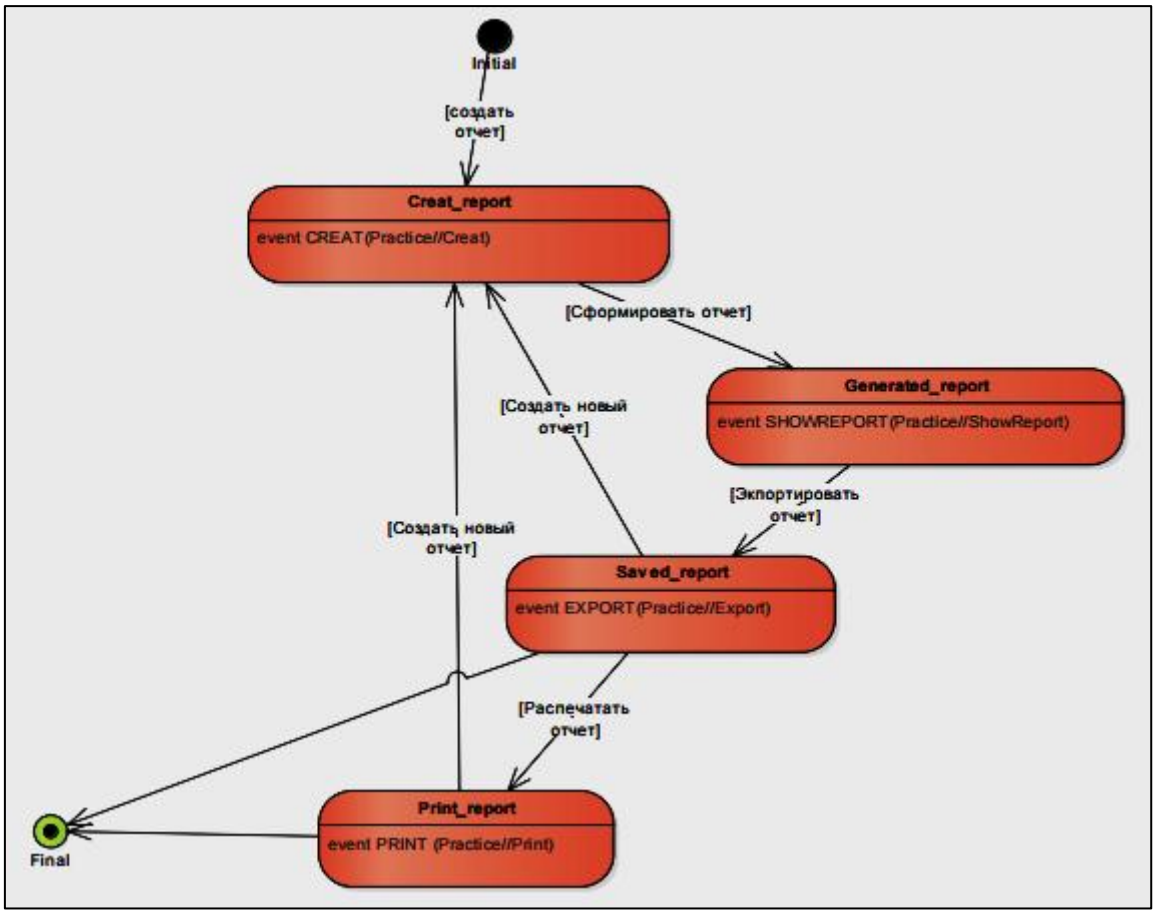


Рисунок 21 — Диаграмма состояний программы

Диаграмма состояний показана на рисунке 21. В ней рассмотрен переход из одного состояния в другое при создании документов.

Переход из состояния «Creat_report» в «Generated_report» происходит после того, как событие CREAT будет выполнено. «Generated_report» это состояние, когда документ сформирован и выведен на экран (событие SHOWREPORT). По желанию пользователя сформированный документ экспортируется в файл формата docx/pdf (событие EXPORT). После чего, также по желанию пользователя, можно направить на печать документ (событие PRINT).

Важно отметить, что из состояний «Saved_report» и «Print_report» можно вернуться в состояние «Creat_report» и создать новый документ.

2.6.6 Диаграммы компонентов и развертывания

Диаграмма компонентов — статическая структурная диаграмма, показывает разбиение программной системы на структурные компоненты и связи (зависимости) между компонентами [14]. На рисунке 22 представлена диаграмма компонентов разрабатываемой системы.

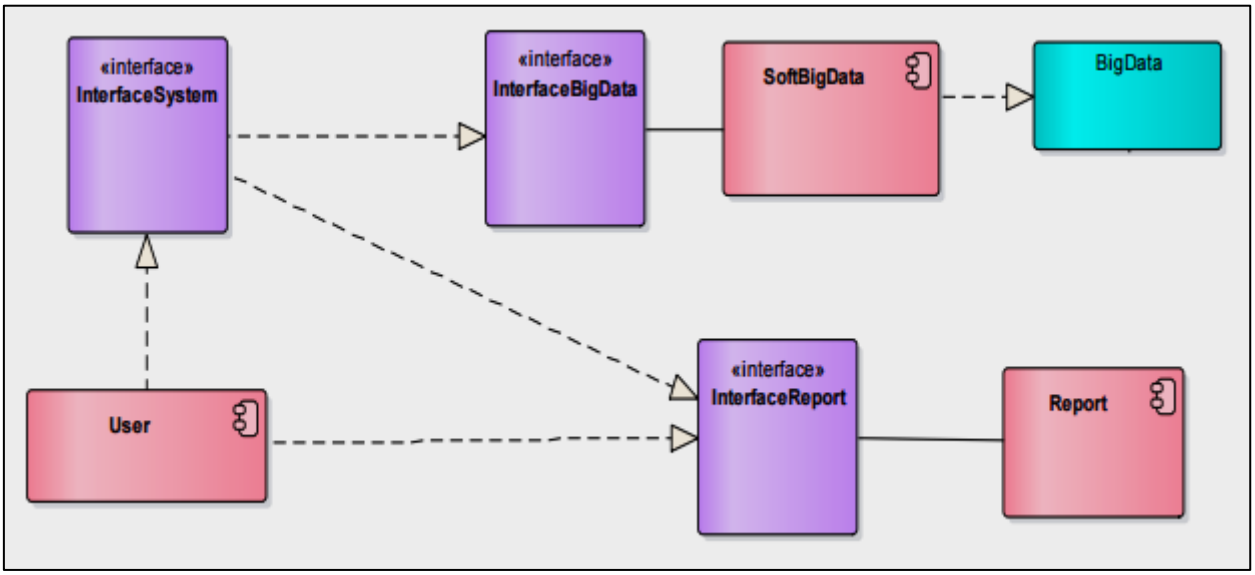


Рисунок 22 — Диаграмма компонентов программы

Элементы диаграммы:

- User — компонент для программного представления пользователя, обеспечивает доступ к БД и отчетам.
- SoftBigData — компонент реализующий взаимодействие с БД системы.
- BigData — база данных.
- Report — компонент работы с отчетами.
- InterfaceSystem — графический интерфейс системы.
- InterfaceBigData — графический интерфейс базы данных.
- InterfaceReport — графический интерфейс при формировании документов.

Между элементами существуют связи: ассоциации (сплошная линия), реализации (пунктирная стрелка). Диаграмма развёртывания UML моделирует физическое развертывание артефактов на узлах.

Архитектура клиент–сервер — это концепция информационной сети, в которой основная часть ее ресурсов сосредоточена в серверах, обслуживающих своих клиентов [15]. Диаграмма развёртывания системы представлена на рисунке 23.

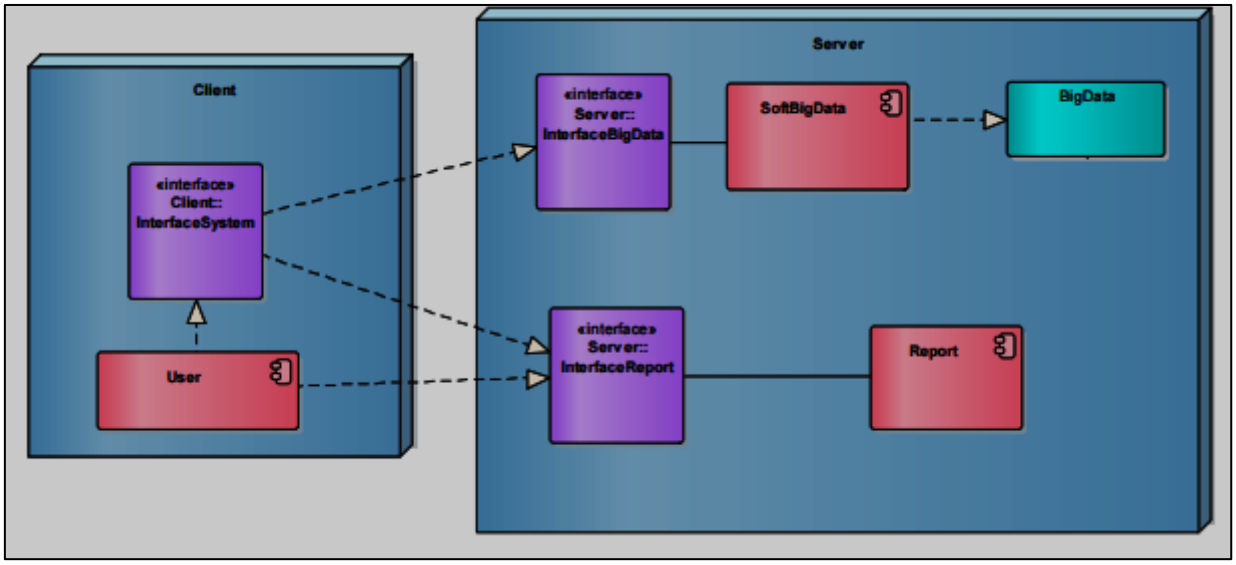


Рисунок 23 — Диаграмма развертывания программы

В отличие от диаграммы компонентов в диаграмме развертывания указано физическое расположение элементов. Все логические связи, представленные на диаграмме компонентов, сохранились и на диаграмме развертывания.

2.7 Вывод по главе 2

В ходе выполнения поставленной задачи проведен анализ требований, предъявляемых в техническом задании (Приложение А) [16]. Анализ сформулированных вариантов использования показал, что с точки зрения потенциальных рисков и архитектурной значимости наиболее существенными являются прецеденты, связанные с работой диспетчера.

Для детализации выбраны два прецедента: «Формирование БД» и «Формирование документов».

В ходе решения поставленной задачи для проектирования модуля АСУ «Web–Деканат» использовались UML–диаграммы. Для разработки программы построено семь видов диаграмм:

- вариантов использования;
- деятельности;
- классов;
- последовательности;
- состояний;
- компонентов;
- развертывания.

Диаграммы отображают, каким образом будет разработан подмодуль АСУ «Web–Деканат».

Глава 3. Разработка подмодуля «Перезачет оценок»

3.1 Прототип разрабатываемой программы

Прототипирование — быстрая реализация базовой функциональности для анализа работы системы в целом. На этапе прототипирования малыми усилиями создается работающая система (возможно неэффективно, с ошибками, и не в полной мере). Во время процесса прототипирования видна более детальная картина устройства системы. Прототипирование, по мнению некоторых разработчиков, является самым важным этапом разработки [17].

Разработанный прототип не удовлетворяет требованию, что программа должна представлять собой модуль АСУ «Web–Деканат» и написана на языке программирования Java. Прототип программы нужен для демонстрирования интерфейса и структуры разрабатываемого модуля. Он не является конечным продуктом, который мог бы использоваться в учебно-организационном отделе инстиута космических и информационных технологий сибирского федерального университета. Также данная программа не выполняет все функции, представленные в техническом задании, а только следующие:

- хранение информации о студентах и предметах;
- редактирование базы данных (добавление, удаление или изменение информации в таблицах БД);
- экспорт сформированного документа в форматы docx/pdf;

Второстепенные функции:

- вход в систему (введение логина и пароля);
- поиск информации по БД;
- печать сформированного документа;
- выход из системы.

Важно отметить, что данная программа разработана только для одного пользователя — диспетчера.

3.2 Архитектура разрабатываемой системы

Архитектура программного обеспечения — совокупность важнейших решений об организации программной системы [18]. На рисунке 24 представлена архитектура АСУ «Web–Деканат». Так как подмодуль «Перезачет оценок» реализуется на базе существующей АИС, то архитектура останется такой же.

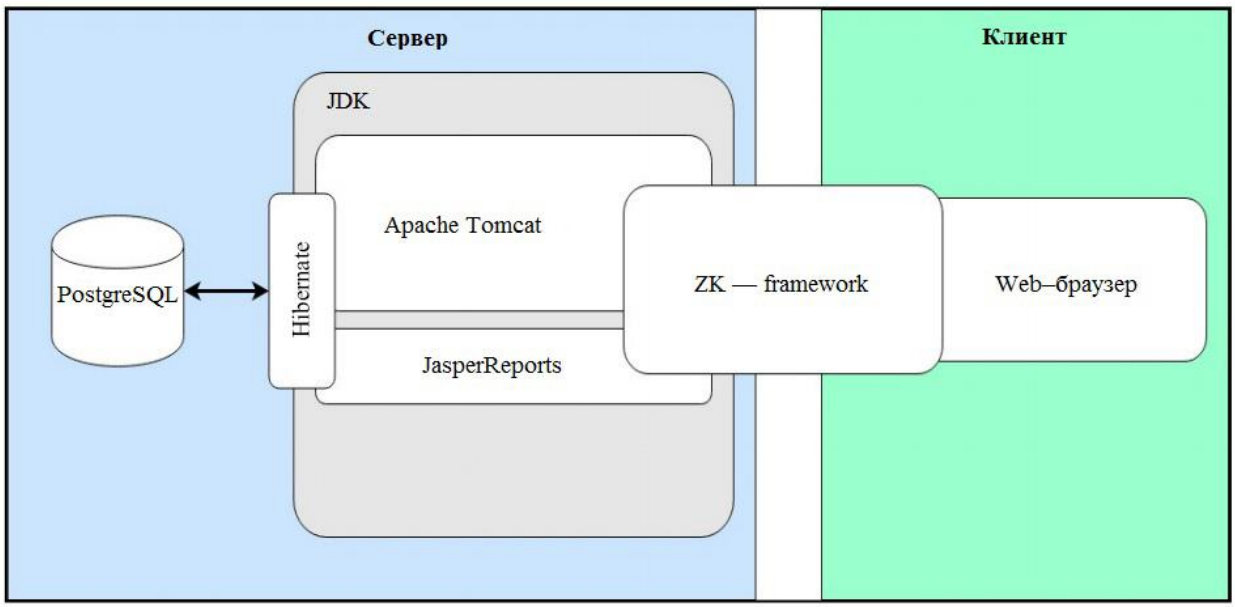


Рисунок 24 — Архитектура разрабатываемого модуля

Модель данных построена в системе управления базами данных (СУБД) PostgreSQL. Hibernate — библиотека для языка программирования Java, предназначенная для решения задач объектно–реляционного отображения. Она представляет собой свободное программное обеспечение с открытым исходным кодом. Данная библиотека предоставляет легкий в использовании каркас (framework) для отображения объектно–ориентированной модели данных в традиционные реляционные базы данных [19].

Java Development Kit (JDK) — бесплатно распространяемый компанией комплект разработчика приложений на языке Java, включающий в себя компилятор Java, стандартные библиотеки классов Java, примеры, документацию, различные утилиты и исполнительную систему Java. В состав JDK не входит интегрированная среда разработки на Java [20]. Для удобства в

создания программного продукта используется интегрированная среда разработки IntelliJ IDEA, которая опирается на сервисы, предоставляемые языком программирования Java Development Kit.

JasperReports — это Java–библиотека для создания отчётов. На основе XML–шаблонов отчётов генерируются готовые для печати документы. В разрабатываемом модуле отчеты выводиться на экран, а после, по желанию пользователя, сохраняется в форматы PDF или DOCX. Основной частью шаблона является SQL–запрос, который указывает какие данные необходимо выбрать из базы данных для генерации отчёта [21].

Apache Tomcat — это контейнер, который позволяет использовать интернет приложения такие, как Java сервлеты и JSP (серверные страницы Java). Сервлет является интерфейсом Java, реализация которого расширяет функциональные возможности сервера. Сервлет взаимодействует с клиентами посредством принципа запрос–ответ [22].

Framework — программная платформа, определяющая структуру программного обеспечения, облегчающее разработку и объединение разных компонентов большого программного проекта [23].

ZK — framework для разработки web–приложений, реализован полностью на Java. Особенность фреймворка — предоставление возможности разработчику создавать полноценные пользовательские интерфейсы для web–приложений [24].

Доступ к подмодулю «Перезачет оценок» АСУ «Web–Деканат» осуществляется с помощью любого web–браузера при наличии доступа к сети Интернет. Также существует возможность работы в системе на компьютерах без подключения к сети Интернет, но только на тех компьютерах, которые соединены по локальной сети СФУ.

Рекомендованные браузеры:

- Google Chrome;
- Mozilla Firefox.

3.3 Модель сущность–связь

Модель сущность–связь (ER–модель) — модель данных, позволяющая описывать концептуальные схемы предметной области. ER–модель используется при высокоуровневом (концептуальном) проектировании баз данных. С её помощью можно выделить ключевые сущности и обозначить связи, которые могут устанавливаться между этими сущностями. Модель сущность–связь реализуется с помощью ER–диаграмм. [25].

Для создания ER–диаграммы применялось программное обеспечение SQL Manager PostgreSQL. Выбор этой программы обусловлен тем, что модель данных построена в СУБД PostgreSQL.

На рисунке 25 представлена сокращенная ER–диаграмма разрабатываемого подмодуля «Перезачет оценок». Полная диаграмма представлена в Приложении Б.

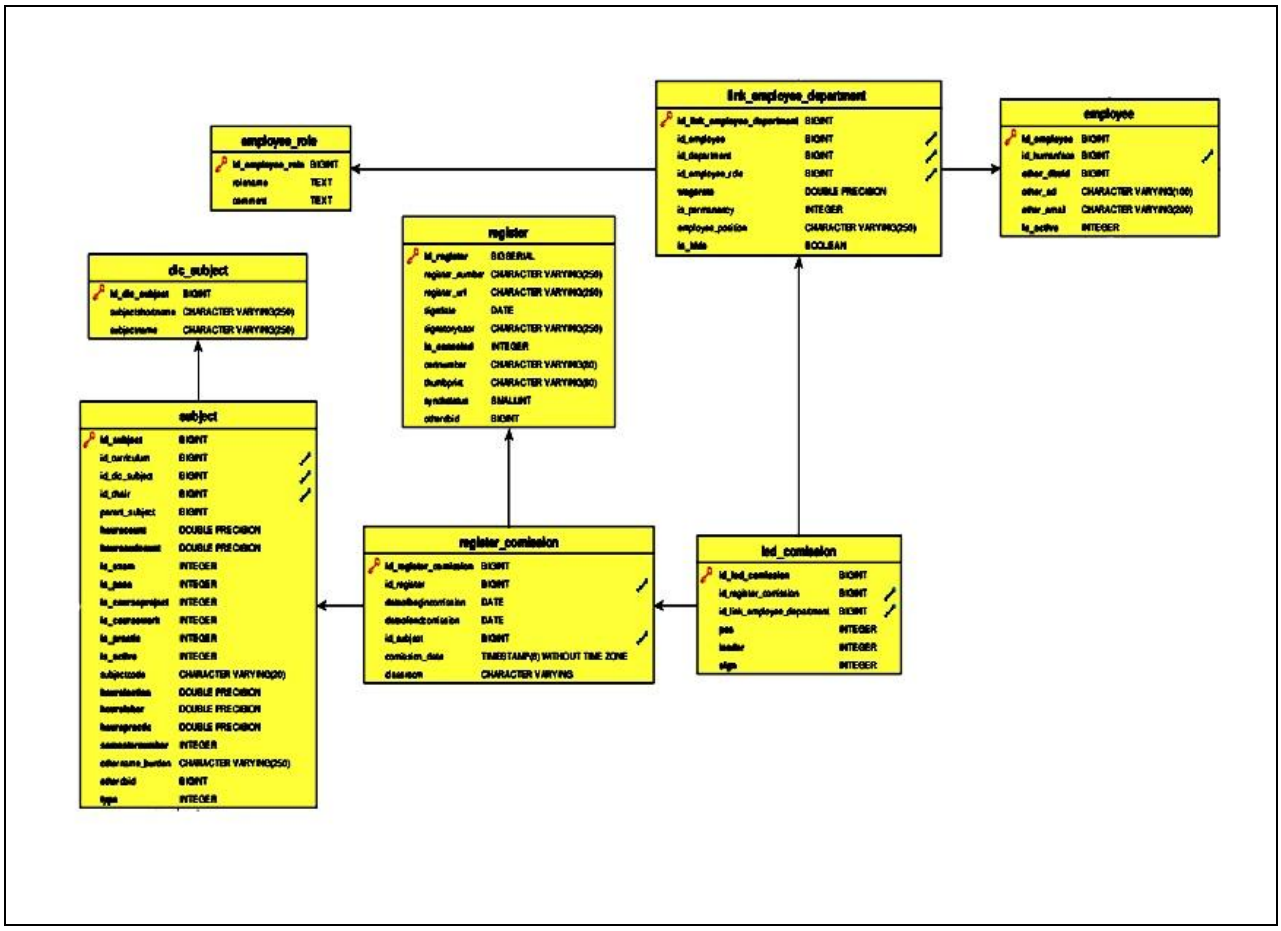


Рисунок 25 — Сокращенная ER–диаграмма подмодуля «Перезачет оценок»

Таблица «subject» показывает данные – дисциплины студента. Таблица «register» показывает оценки обучающихся в институте космическх и информационных технологий сибирского федерального университета. Она соединена с другой таблицей по ключевому полю «id_register» (связь «один ко многим»).

3.4 Основное меню подмодуля «Перезачет оценок»

После процедуры авторизации в автоматизированной системе управления «Web–Деканат» диспетчер учебно-организационного отдела института космических и информационных технологий сибирского федерального университета наблюдает основное меню системы, которое представлено на рисунке 26.

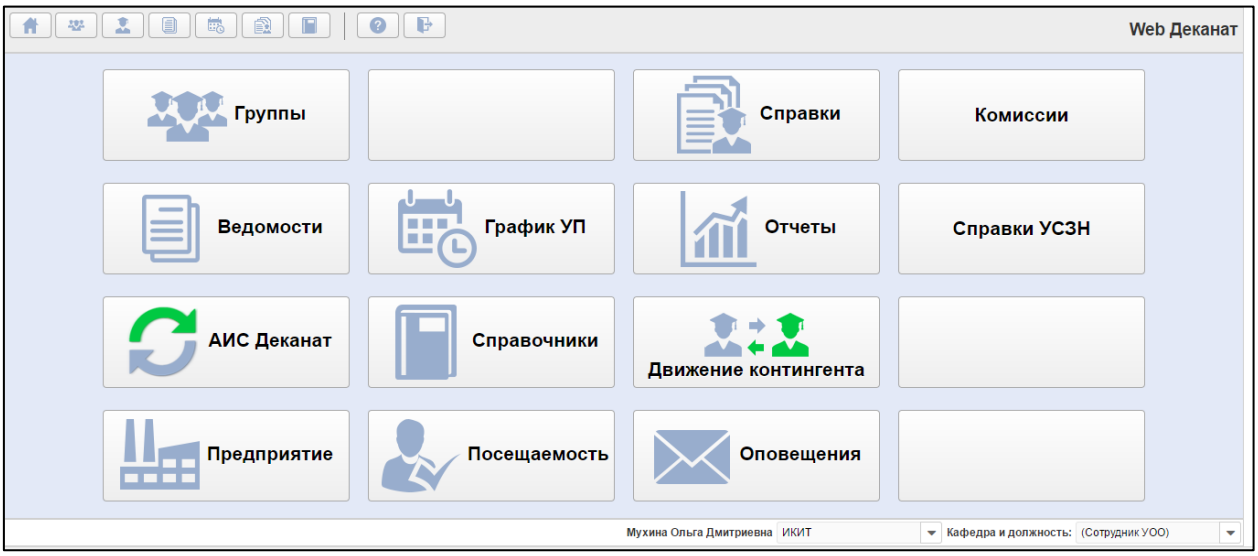


Рисунок 26 — Основное меню АСУ «Web–Деканат»

Справа рисунка расположена кнопка «Движение контингента». Важно отметить, что дополнительная авторизация к разрабатываемому подмодулю не требуется.

На рисунке 27 представлено основное меню модуля «Движение контингента» АСУ «Web–Деканат».

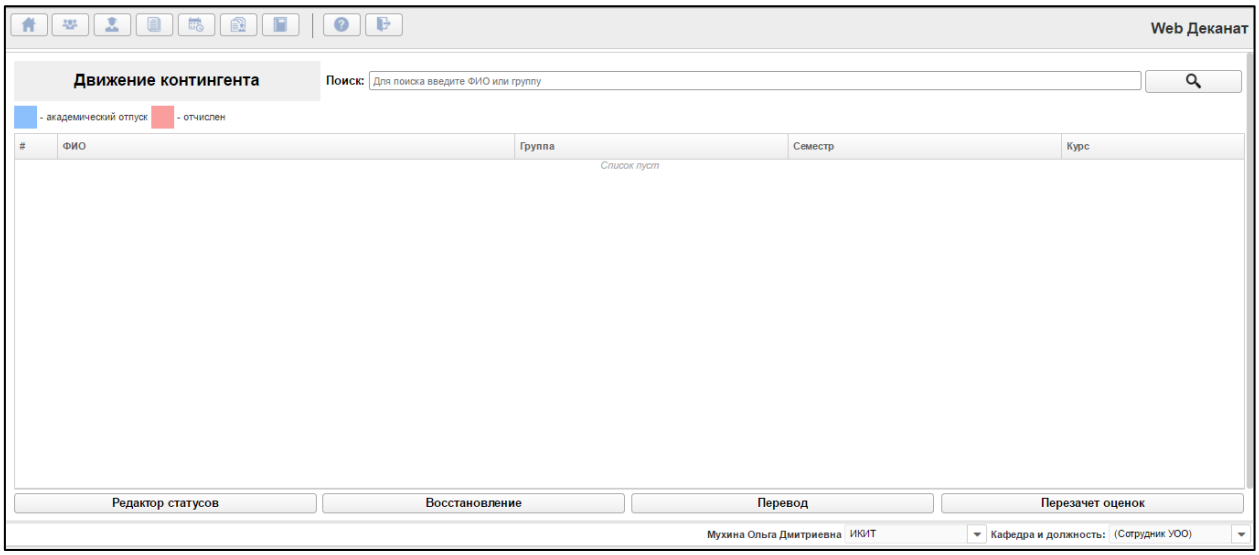


Рисунок 27 — Основное меню модуля «Движение контингента» АСУ «Web–Деканат»

Интерфейс меню реализован четырьмя кнопками: «Редактор статусов», «Восстановление», «Перевод» и «Перезачет оценок». Каждая кнопка представляет свой подмодуль.

Для того чтобы определить учебный план, с которым будет происходить сравнение, необходимо восстановить заявителя в число студентов и определить его в нужную группу – рисунок 28.

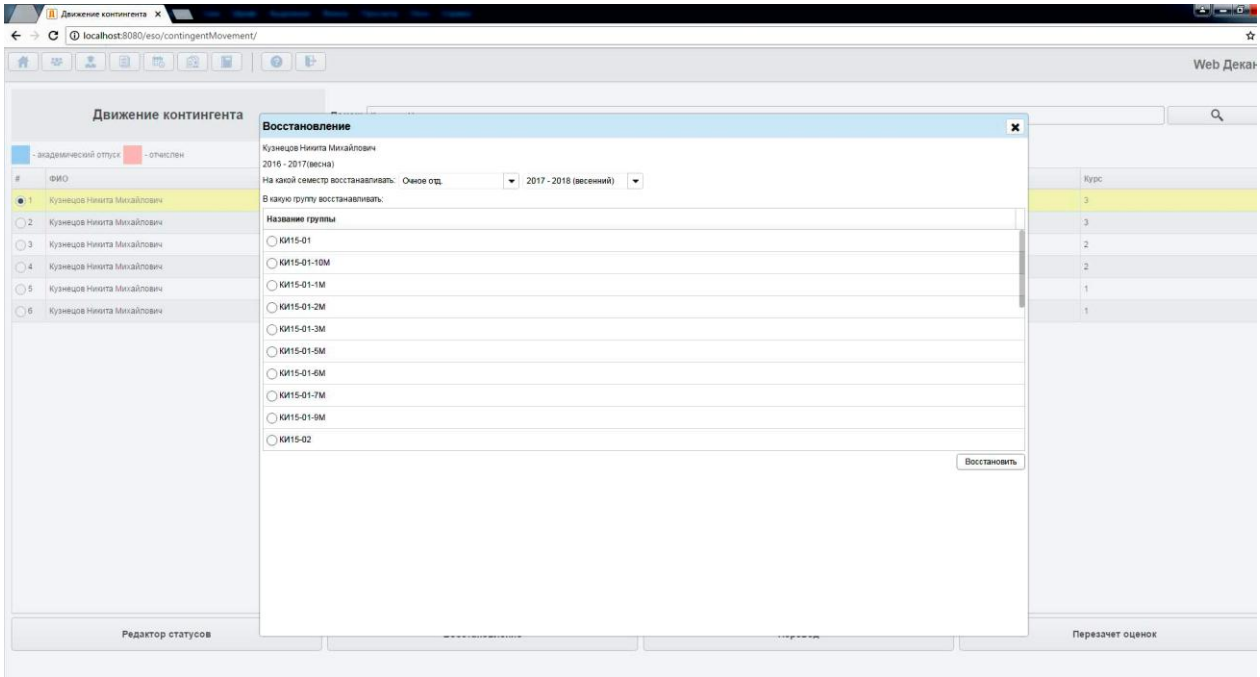


Рисунок 28 — Подмодуль «Восстановление»

3.5 Подмодуль «Перезачет оценок»

На рисунке 29 представлен интерфейс подмодуля «Перезачет оценок». Подмодуль реализован в табличной форме с указанием наименования полей, где хранится некоторая информация об учебном плане и студенте.

Для удобства пользователя в использовании подмодуля основные данные структурированы и представлены в виде списка от 1 семестра к последнему.

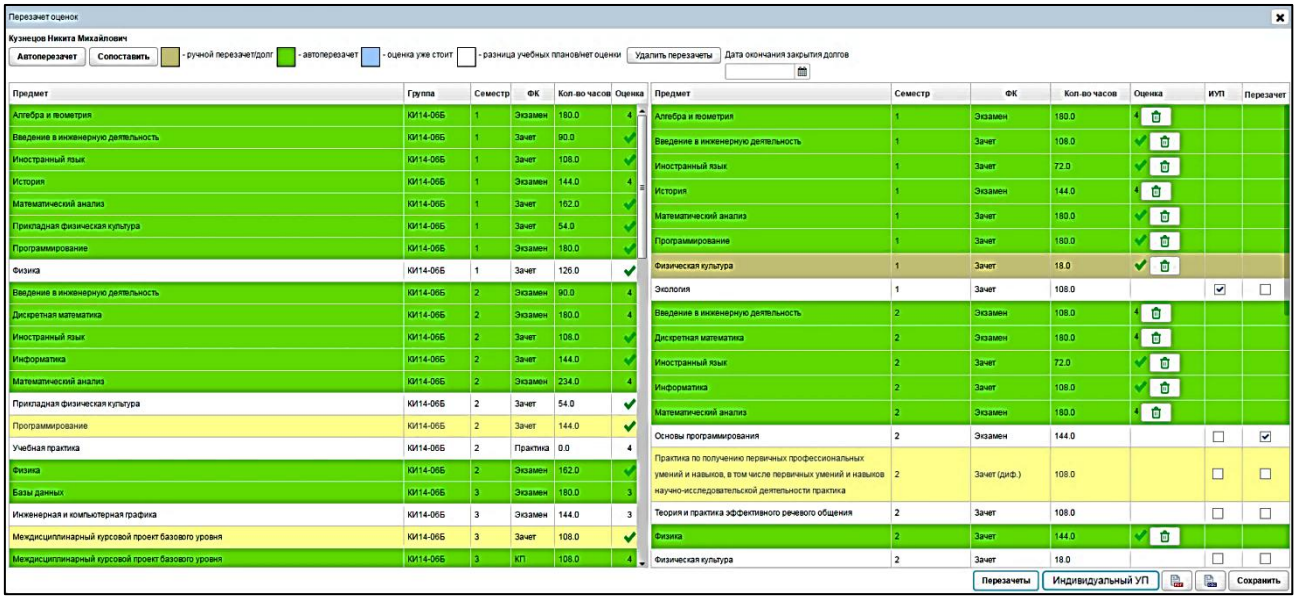


Рисунок 29 — Подмодуль «Перезачет оценок»

Диспетчер может воспользоваться кнопкой автоперезачет или произвести сверку учебных планов вручную. На рисунке 29, показан интерфейс ручного и автоматического перезачета дисциплин. Диспетчер определяет предмет в правой таблице, затем в левой таблице находит предмет который соответствует предмету в правой таблице и нажимает кнопку «Сопоставить» либо пользуется кнопкой «Автоперезачет».

Важно отметить, что правильность заполнения данных в данном подмодуле зависит от диспетчера УОО ИКИТ СФУ. В случае ошибки при сопоставлении формы контроля дисциплин программа выдаст уведомление о невозможности сопоставления.

После добавления или изменения информации о студенте, данные мгновенно обновляются в паспортах групп, а так же автоматически открываются электронные ведомости.

3.6 Формирование документов

На основании внесенных данных диспетчером УОО ИКИТ СФУ формируются документы:

– Индивидуальный учебный план;

– Ведомость перезачета дисплин.

На рисунке 30 представлен интерфейс подмодуля «Перезачет оценок». На данном рисунке отображен список документов, которые в автоматизированном режиме можно сформировать. В дальнейшем количество типов документов, формируемых подмодулем «Перезачет оценок» АСУ «Web–Деканат», может вырасти, в связи с какими–либо изменениями в положении о переводе и восстановление в число студентов СФУ.

Рисунок 30 — Печать и сохранение документов

Документы формируются по определенным заранее подготовленным стандартным шаблонам. Данные подгружаются в шаблоны автоматически, на рисунке 31 представлена сформированная ведомость перезачета дисциплин, в которой представлены дисциплины перезачтенные полностью и которые не нуждаются в согласовании с преподавателями.

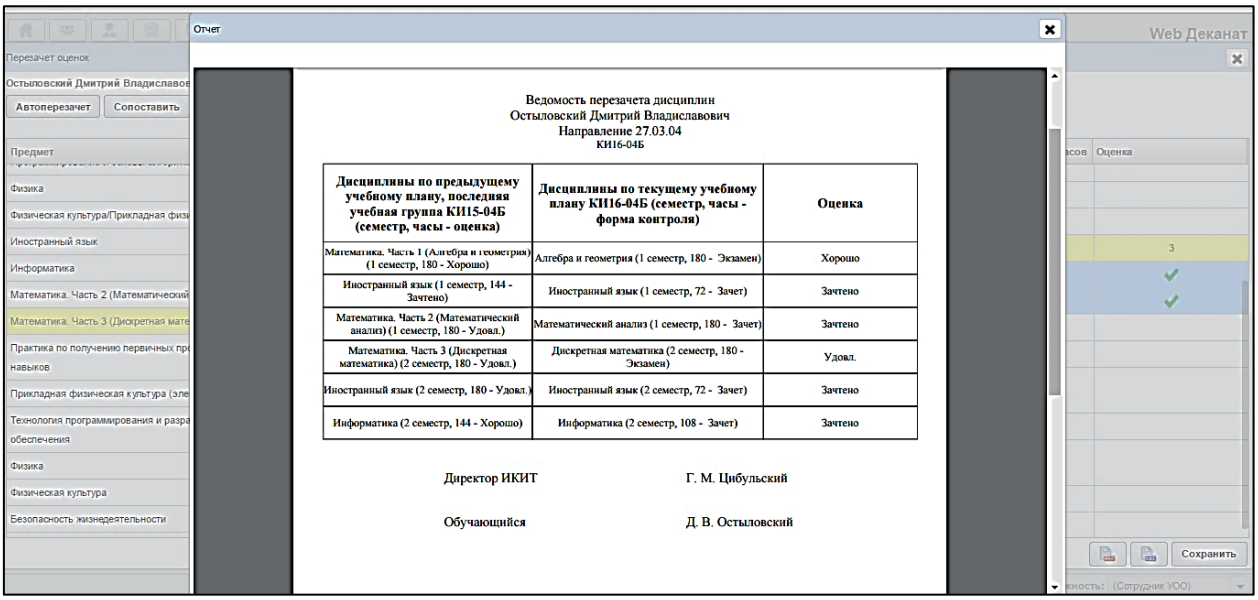


Рисунок 31 — Документ «Ведомость перезачета дисциплин»

На рисунке 32 представлен сформированный индивидуальный учебный план студента.

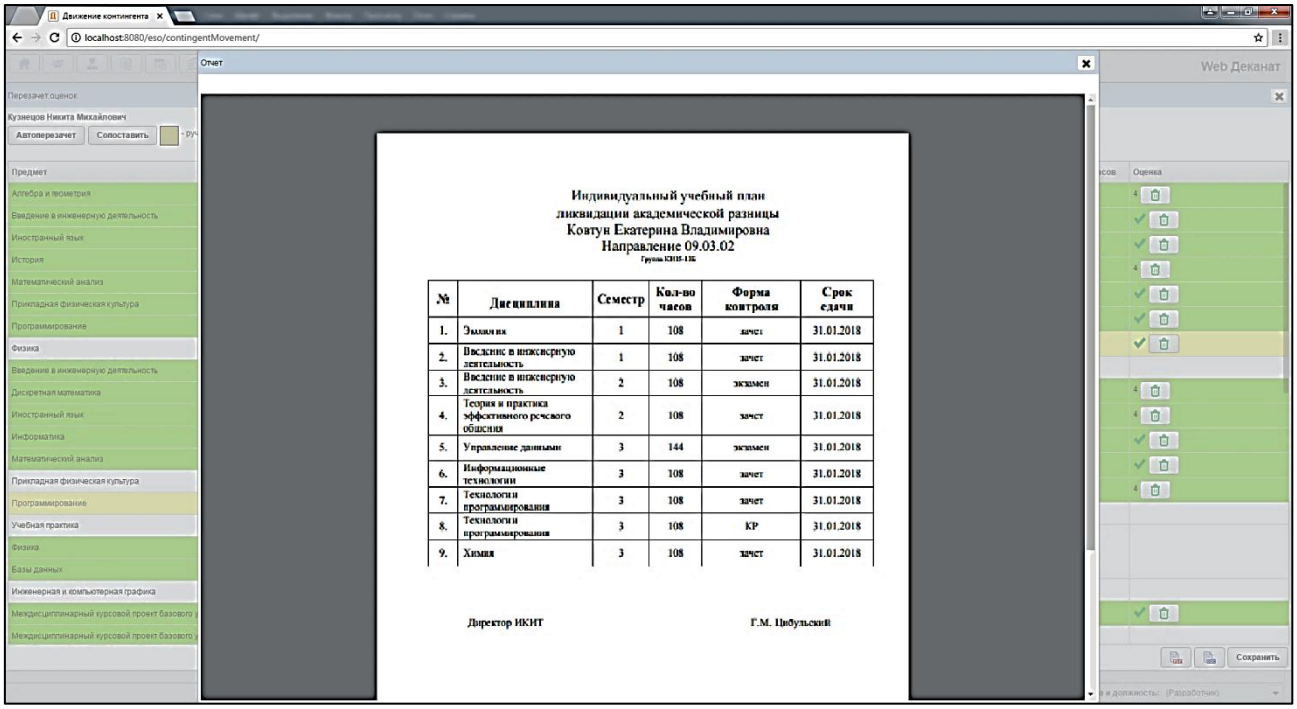


Рисунок 32 — Документ «Индивидуальный учебный план»

Остальные документы формируются аналогичным образом.

После формирования документов по желанию диспетчера их можно сохранить в электронном виде или распечатать на бумажный носитель.

Для того, чтобы уточнить в паспорте групп есть ли у студента разница в учебных планах, можно вывести подробную информацию по определенному студенту. Пример представлен на рисунке 33.

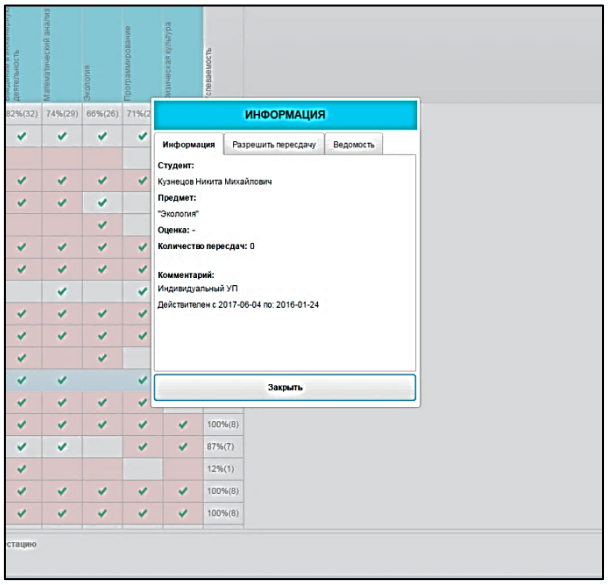


Рисунок 33 — Информация в паспорте групп о наличие ИУП

3.7 Вывод по главе 3

В ходе выполнения поставленной задачи разработан прототип программы.

Он не удовлетворяет требованию, что программа должна представлять собой подмодуль АСУ «Web–Деканат» и написана на языке Java. Прототип программы нужен для демонстрирования интерфейса и структуры разрабатываемого подмодуля.

Подмодуль «Перезачет оценок» АСУ «Web–Деканат» разработан и на данный момент функционирует в тестовом режиме. Система реализована на языке программирования Java. Модель данных построена в СУБД PostgreSQL. Для удобства создания программного продукта используется интегрированная среда разработки IntelliJ IDEA, которая опирается на сервисы, предоставляемые JDK.

Библиотека JasperReports использовалась для формирования документов для установления разницы учебных планов.

Отличительная особенность системы в том, что она доступна в любое время суток, на любом компьютере, где есть подключение к сети Интернет.

В третьей главе подробно представлена работа подмодуля «Перезачет оценок» АСУ «Web–Деканат». Все требования, согласно техническому заданию, были выполнены.

ЗАКЛЮЧЕНИЕ

В ходе выполнения бакалаврской работы рассмотрена деятельность диспетчера УОО ИКИТ, ответственного за установку и подготовку документов по разнице учебных планов. Выделены следующие проблемы при работе диспетчера:

- Не автоматизирован процесс сопоставления дисциплин.

- Создание различных документов проходит с помощью пакета Microsoft Office. Это трудоемкий процесс со значительной вероятностью ошибки в сравнении с использованием АИС для конкретных задач установления разницы учебных планов.

Для того, чтобы решить выявленные проблемы, требуется создать подмодуль «Перезачет оценок» для существующей АСУ «Web–Деканат». Нецелесообразно создавать отдельное программное обеспечение для нужд диспетчера УОО ИКИТ СФУ. Так как, действующая АСУ имеет базу студентов и учебных планов нужную для формирования необходимых документов.

Анализ сформулированных вариантов использования для разрабатываемого подмодуля показал, что с точки зрения потенциальных рисков и архитектурной значимости наиболее существенными являются прецеденты, связанные с работой диспетчера.

Для дальнейшей детализации выбраны два прецедента: «Формирование БД», «Формирование документов».

Для проектирования разрабатываемой программы использовались UML–диаграммы. Для разработки программы построено семь видов диаграмм, представленных в бакалаврской работе. Диаграммы отображают, каким образом будет работать подмодуль «Перезачет оценок» АСУ «Web–Деканат».

Для разработки подмодуля использовался такой метод выявления требований, как прототипирование. Он помог определить структуру, уточнить функции и определиться с интерфейсом разрабатываемой программы.

Пододуль «Презачет оценок» АСУ «Web–Деканат» разработан и на данный момент функционирует в тестовом режиме.

Система реализована на языке программирования Java. Модель данных построена в СУБД PostgreSQL. Для удобства создания программного продукта используется интегрированная среда разработки IntelliJ IDEA, которая опирается на сервисы, предоставляемые JDK. Библиотека JasperReports использовалась для создания отчетов по практике.

Отличительная особенность системы в том, что она доступна в любое время суток, на любом компьютере, где есть подключение к сети Интернет.

Во второй главе подробно представлена работа подмодуля «Перезачет оценок» АСУ «Web–Деканат». Все требования, согласно техническому заданию (Приложение А) [9], были выполнены.

Стабильная работа системы планируется с 1 сентября 2017 года.

Статья по теме данной работе будет опубликована в XLVI Студенческой международной заочной научно–практической конференции «Молодежный научный форум: технические и математические науки».

Подмодуль «Перезачет оценок» АСУ «Web–Деканат» разработан при выполнении бакалаврской работы по направлению 09.03.02 «Информационные системы и технологии», по профилю подготовки 09.03.02.05 «Информационные системы и технологии в административном управлении».

СПИСОК ИСПОЛЬЗОВАННЫХ ИСТОЧНИКОВ

1) Приложение к распоряжению №10 от 20.01.2015 «График выполнения работ по представления документов для организации учебного процесса в СФУ»

2) Информационная система "Деканат" [Электронный ресурс], - Режим доступа: http://www.mmis.ru/programs/dekanat

3) Программный комплекс "Электронные ведомости" [Электронный ресурс], - Режим доступа: http://www.mmis.ru/programs/vedkaf

4) 1С:Университет [Электронный ресурс], - Режим доступа: http://solutions.1c.ru/catalog/university/features

5) Ботнаренко, И.В., Кротов, Я.К., Куренков, И.С., Терентьев, Д.С., Система накопления знаний (Электронный деканат) [Статья] // Электронные средства и системы управления. – 2014. – № 2. – С. 31–33. – УДК: 621.396.41

6) Иванилова, Т.Н, Буслов, И. А., Система электронной поддержки деятельности деканата [Статья] // Современные тенденции развития педагогических технологий в медицинском образовании. Вузовская педагогика. – 2016. – С. 428–440. – УДК: 378:61(063)

7) Положение о порядке перевода, восстановления и отчисления студентов Федерального государственного автономного образовательного учреждения высшего профессионального образования «Сибирский федеральный университет» [Электронный ресурс], - Режим доступа: http://about.sfu–kras.ru/docs/8094/pdf/360667

8) Статья «UML» [Электронный ресурс], - Режим доступа: http://www.omg.org/spec/UML/2.5/

9) Статья «Диаграмма прецедентов» [Электронный ресурс], - Режим доступа: http://www.intuit.ru/studies/courses/1007/229/lecture/5962/

10) Статья «Диаграмма деятельности» [Электронный ресурс], - Режим доступа: http://it-gost.ru/articles/view_articles/96

11) Статья «Диаграмма классов» [Электронный ресурс], - Режим доступа: http://holub.com/uml/

12) Статья «Диаграмма последовательности» [Электронный ресурс], - Режим доступа: https://www.ibm.com/developerworks/rational/library/3101.html

13) Статья «Диаграмма состояний» [Электронный ресурс], - Режим доступа: http://www.agilemodeling.com/artifacts/sequenceDiagram.htm

14) Статья «Диаграмма компонентов» [Электронный ресурс], - Режим доступа: http://www.omg.org/spec/UML/2.1.2/Superstructure/PDF

15) Статья «Диаграмма развертывания» [Электронный ресурс], - Режим доступа: http://www.agilemodeling.com/artifacts/deploymentDiagram.htm

16) ГОСТ 19.201–78 (ЕСПД) Техническое задание. Требования к содержанию и оформлению.

17) Статья «Прототипирование» [Электронный ресурс], - Режим доступа: http://www.4living.ru/items/article/rapid-prototyping/

18) Статья «Архитектура ПО» [Электронный ресурс], - Режим доступа: https://msdn.microsoft.com/ru–ru/hh144976.aspx

19) Статья «Hibernate (библиотека)» [Электронный ресурс], - Режим доступа: http://docs.jboss.org/hibernate/annotations/3.5/reference/en/html_single/

20) Статья «Java Development Kit» [Электронный ресурс], - Режим доступа: http://www.oracle.com/technetwork/java/javase/downloads/jdk8-downloads-2133151.html

21) Статья «JasperReports» [Электронный ресурс], - Режим доступа: http://community.jaspersoft.com/project/jasperreports-library

22) Статья «Apache Tomcat» [Электронный ресурс], - Режим доступа: https://tomcat.apache.org/#Tomcat_8.5.11_Released

23) Статья «Фреймворк» [Электронный ресурс], - Режим доступа: http://citforum.ru/SE/project/isr/

24) Статья «ZK» [Электронный ресурс], - Режим доступа: https://www.ibm.com/developerworks/library/wa-aj-zkquery/

25) Статья «ER–модель данных» [Электронный ресурс], - Режим доступа: http://citforum.ru/database/classics/chen/

26) Постановление Министерства труда и социального развития РФ от 01.01.01г. «Об утверждении межотраслевых типовых норм времени на работы по сервисному обслуживанию ПЭВМ и оргтехники и сопровождению программных средств».

27) ГОСТ Р 50948–2001. Средства отображения информации индивидуального пользования. Общие эргономические требования и требования безопасности.

28) ГОСТ Р 50923–96. Дисплеи. Рабочее место оператора. Общие эргономические требования и требования к производственной среде.

29) ГОСТ 21889–76 Система "Человек–машина". Кресло человека–оператора. Общие эргономические требования.

30) ГОСТ 12.1.005–88 ССБТ. Общие санитарно–гигиенически требования к воздуху рабочей зоны (с Изменением N 1).

ПРИЛОЖЕНИЕ А

Техническое задание

Наименование — «Автоматизированное рабочее место диспетчера учебно–организационного отдела института космических и информационных технологий». Программа предназначена для упрощения документооборота по формированию приказов на перевод, восстановление и выход из академического отпуска обучающихся. Предполагает создание автоматизированной сверки учебных планов и формирование индивидуальных планов и ведомостей.

1 Основания для разработки

Основанием для проведения разработки программы является положение о порядке перевода, восстановления и отчисления студентов Федерального государственного автономного образовательного учреждения высшего профессионального образования «Сибирский федеральный университет», принятое на заседании ученого совета – протокол №1 от 27.01.2014 [7]. Также основанием является поручение заместителя директора по учебной работе ИКИТ Ю. Ю. Якунина. Наименование — «Разработка автоматизированного рабочего места диспетчера УОО ИКИТ СФУ». Условное обозначение — «Перезачет оценок».

2 Назначение разработки

Функциональным назначением программы является предоставление пользователю возможности работы с базой данных учебных планов, студентов и формирования соответствующей отчетности. Программа эксплуатируется в учебно–организационном отделе (УОО) ИКИТ. Конечным пользователем является диспетчер УОО ИКИТ СФУ.

3 Требования к программе

Программа обеспечивает выполнение следующих основных функций:

- хранение информации о студентах и учебных планах;

- редактирование базы данных (добавление, удаление или изменение информации);

- формирование отчетности (разница в учебных планах, индивидуальный учебный план (план–график) ликвидации академической разницы, индивидуальная ведомость ликвидации академической разницы по индивидуальному учебному плану (плану–графику), ведомость перезачета дисциплин, протокол аттестационной комиссии), путем внесения информации из БД, в имеющиеся шаблоны документов;

- экспорт сформированного отчета в форматы .docx, .pdf.

Второстепенные функции:

- вход в систему (ввод логина и пароля);

- поиск информации по БД;

- отображение инструкции пользователя;

- печать сформированного отчета;

- выход из системы.

Входными данными считается внесение информации в БД. Для того чтобы выбрать необходимые учебные планы для установления разницы при открытии программы будет предложен выбор с помощью выпадающих окон: студента и его предыдущих учебных планов (автоматизировано), а так же выбор будущего учебного плана. После выбора данные будут представлены в виде таблицы. В таблице можно сравнить данные. В программе должно быть две таблицы: «Учебный план «А»», «Учебный план «В»» и кнопка «Автоперезачет» и «Сопоставить». Неперезачтенные дисциплины помечаются дополнительно в зависимости от того к какой категории относятся: ИУП или перезачеты. Поля таблицы «Учебный план «А»»:

- Предмет;

- Группа;
- Семестр;
- Форма контроля (ФК);
- Количество часов;
- Оценка.

Следует отметить, что разрабатываемая система, является модулем АСУ «Web–Деканат» и поэтому данные о студентах имеются. В связи с этим создавать такую таблицу в БД не требуется. Поля таблицы «Учебный план «В»»:

- Предмет;
- Семестр;
- Форма контроля (ФК);
- Количество часов;
- Оценка;
- ИУП;
- Перезачет.

Алгоритм перезачетов выглядит следующим образом:

1. Если форма контроля была «зачет», а стала «экзамен», то дисциплина предлагается на перезачет.
2. Если названия дисциплин и форма контроля совпадают, но разница в количестве часов более чем 30%, то дисциплина предлагается на перезачет.

Выходные данные программы представляют собой сформированный отчет в виде документа формата .docx и .pdf. Требования к временным характеристикам предъявляются в выпадающем окне.

3.1 Требования к надежности

Надежное (устойчивое) функционирование программы обеспечено выполнением Заказчиком совокупности организационно–технических мероприятий, перечень которых приведен ниже:

- организация бесперебойного питания технических средств;

- использование лицензионного программного обеспечения;

- регулярное выполнение рекомендаций Министерства труда и социального развития РФ, изложенных в Постановлении от 01.01.01 г. «Об утверждении межотраслевых типовых норм времени на работы по сервисному обслуживанию ПЭВМ и оргтехники и сопровождению программных средств» [26];

- использование антивирусных программ;

- регулярное выполнение требований ГОСТ Р 50923–96 [28], ГОСТ Р 50948–2001 [27].

Критерии надежности программы:

- время, затрачиваемое на обслуживание системы, не должно превышать 3 % от общего времени работы;

- среднее время безотказной работы десять рабочих дней;

- максимальная норма ошибок или дефектов — одна ошибка на десять тысяч строк кода.

Время восстановления после отказа, вызванного сбоем электропитания технических средств (иными внешними факторами), не фатальным сбоем 62 операционной системы, не должно превышать 15 минут при условии соблюдения условий эксплуатации технических и программных средств. Время восстановления после отказа, вызванного неисправностью технических средств, фатальным сбоем операционной системы, не должно превышать времени, требуемого на устранение неисправностей технических средств и переустановки программных средств.

3.2 Условия эксплуатации

Климатические условия эксплуатации соответствуют ГОСТ 21889–76 [29], ГОСТ 12.1.005–88 [30]. Обязательное выполнение Заказчиком совокупности организационно– технических мероприятий, перечень которых приведен ниже:

- организация бесперебойного питания технических средств;

- использование лицензионного программного обеспечения;

- регулярное выполнение рекомендаций Министерства труда и социального развития РФ, изложенных в Постановлении от 01.01.01 г. «Об утверждении межотраслевых типовых норм времени на работы по сервисному обслуживанию ПЭВМ и оргтехники и сопровождению программных средств» [26];

- использование антивирусных программ;

- регулярное выполнение требований ГОСТ Р 50923–96 [28], ГОСТ Р 50948–2001 [27].

Минимальное количество персонала, требуемого для работы программы, должно составлять не менее 2 штатных единиц — системный администратор и конечные пользователи программы: диспетчер, сотрудник кафедры. Системный администратор должен иметь высшее профильное образование в сфере информационных систем и технологий. В перечень задач, выполняемых системным администратором, должны входить:

- задача поддержания работоспособности технических средств;

- задачи установки (инсталляции) и поддержания работоспособности системных программных средств операционной системы;

- задача установки (инсталляции) программы.

Конечный пользователь программы должен обладать практическими навыками работы с графическим пользовательским интерфейсом операционной системы. Базируется программа на рабочем стационарном компьютере диспетчера. Минимальные системные требования:

- Процессор Pentium с частотой не менее 300 МГц.

- Не менее 512 МБ оперативной памяти.

- Не менее 1,5 ГБ свободного места на жестком диске.

- Клавиатура, компьютерная мышь.

- Видеокарта и монитор, поддерживающие режим Super VGA с разрешением не менее чем 800 на 600 точек

- Звуковая плата
- Динамики
- Операционная система Windows XP и выше.

3.3 Требования к информационной и программной совместимости

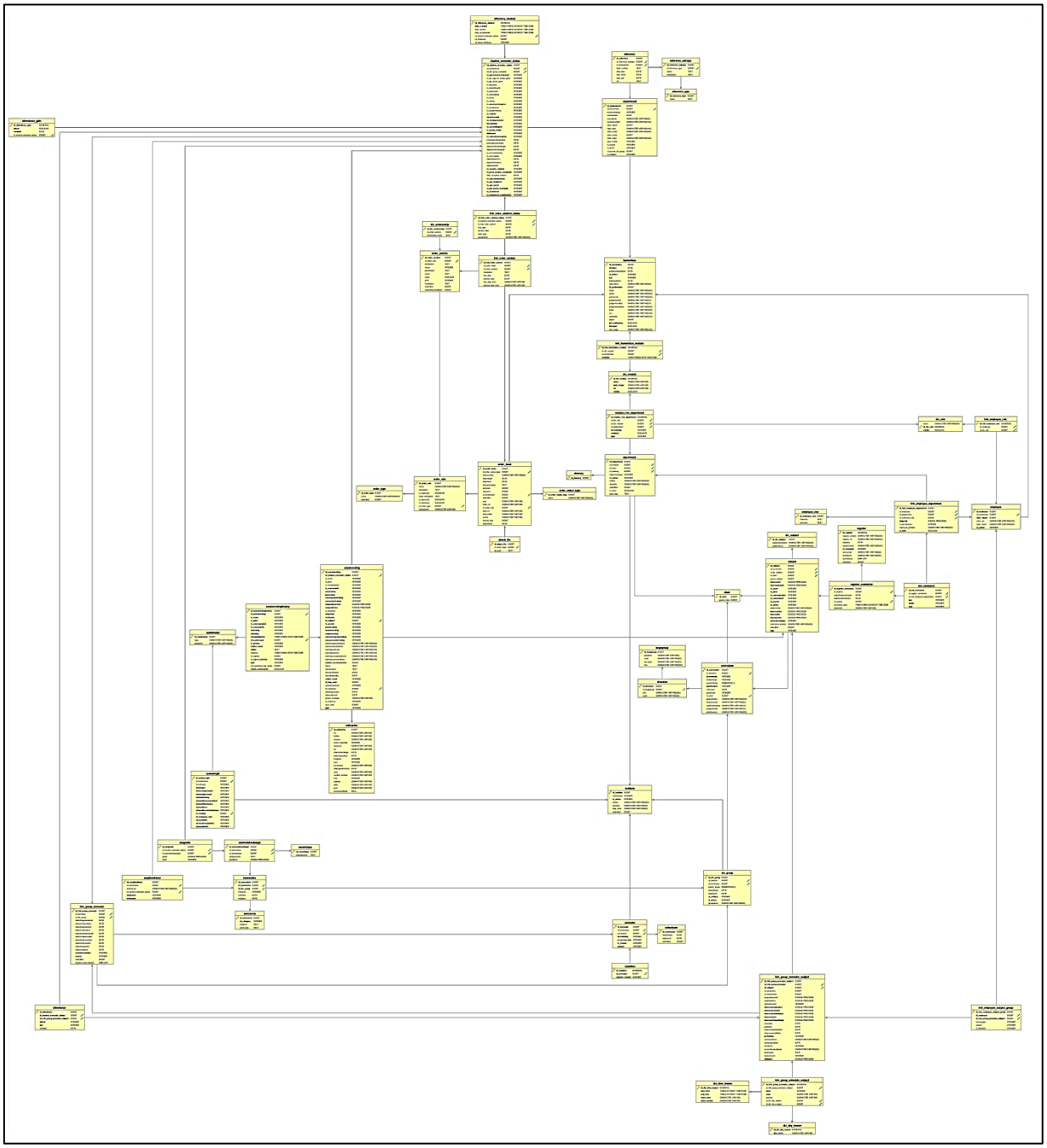
К информационной структуре входных данных требования не предъявляются. Информационная структура файлов на выходе должна соответствовать шаблонам заданных отчетов. Информационная структура включает в себя текст, содержащий разметку, предусмотренную спецификацией формата docx. Требования к программным средствам, используемым программой. Системные программные средства, используемые программой, должны быть представлены лицензионной локализованной версией операционной системы. 64 Требования к защите информации и программ. Доступ к данным, хранящимся в БД, имеют только диспетчеры и ответственные за практику на кафедрах. Диспетчер имеет полный функциональный доступ в АИС. Сотрудник кафедры имеет доступ к формированию отчетов, просмотру БД, но не редактировании ее.

3.4 Специальные требования

Программа обеспечивает взаимодействие с пользователем посредством графического пользовательского интерфейса, разработанного согласно уже имеющийся АСУ «Web–Деканат».

ПРИЛОЖЕНИЕ Б

Диаграмма сущность–связь подмодуля «Перезачет оценок»



ПРИЛОЖЕНИЕ В

Плакаты презентации

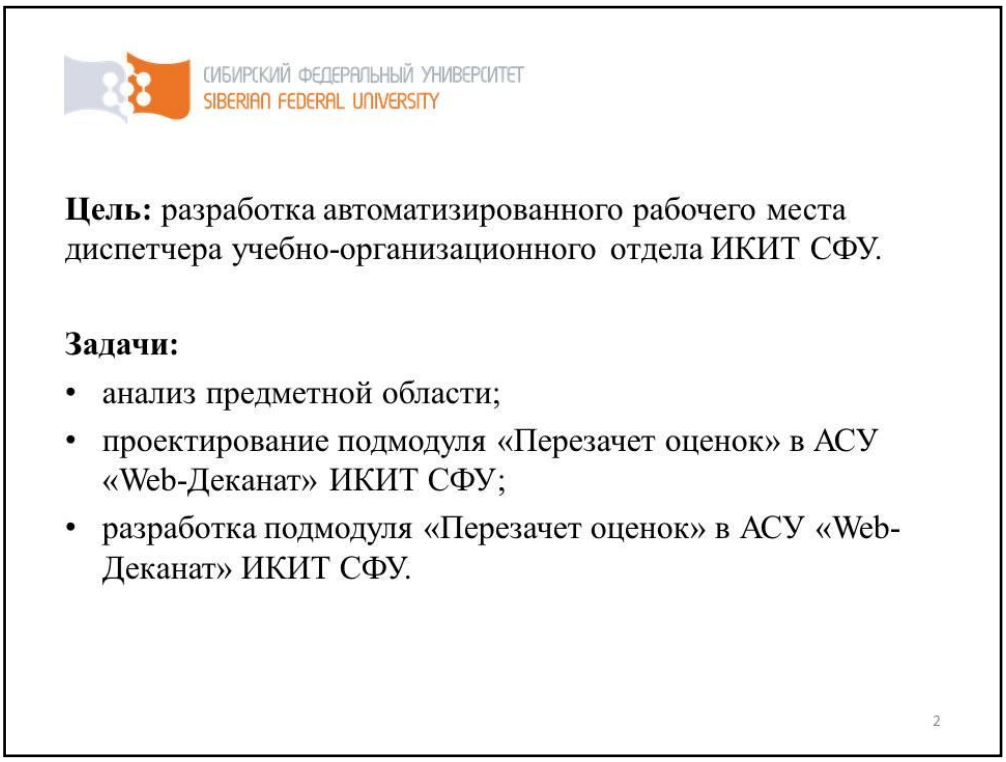


Рисунок В.1 – Слайд презентации № 1

Рисунок В.2 – Слайд презентации № 2

СИБИРСКИЙ ФЕДЕРАЛЬНЫЙ УНИВЕРСИТЕТ
SIBERIAN FEDERAL UNIVERSITY

Актуальность

- Применение информационных технологий позволяют автоматизировать задачи создания условий для свободного функционирования и развития системы высшего образвания (задача Министерства образования);
- На текущий момент в УОО ИКИТ используется АСУ «Web-Деканат», но в ней не реализованы функции диспетчера УОО ответственного за установление разницы в учебных планах;
- Разрабатываемая программа является модулем существующей АСУ «Web-Деканат».

3

Рисунок В.3 – Слайд презентации № 3

СИБИРСКИЙ ФЕДЕРАЛЬНЫЙ УНИВЕРСИТЕТ
SIBERIAN FEDERAL UNIVERSITY

Анализ предметной области

- «1С:Университет» (модуль «Управление контингентом»)
- «Деканат» лаборатории ММИС (г. Шахты)
- «Электронный деканат» реализованный на СДО Moodle

4

Рисунок В.4 – Слайд презентации № 4

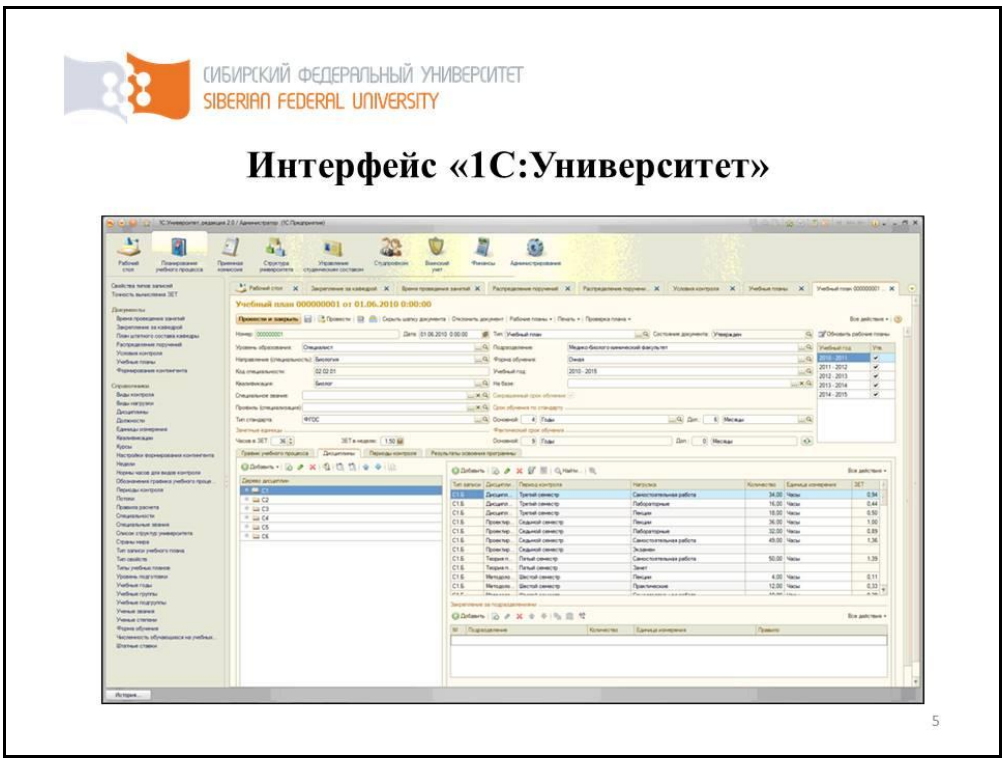


Рисунок В.5 – Слайд презентации № 5

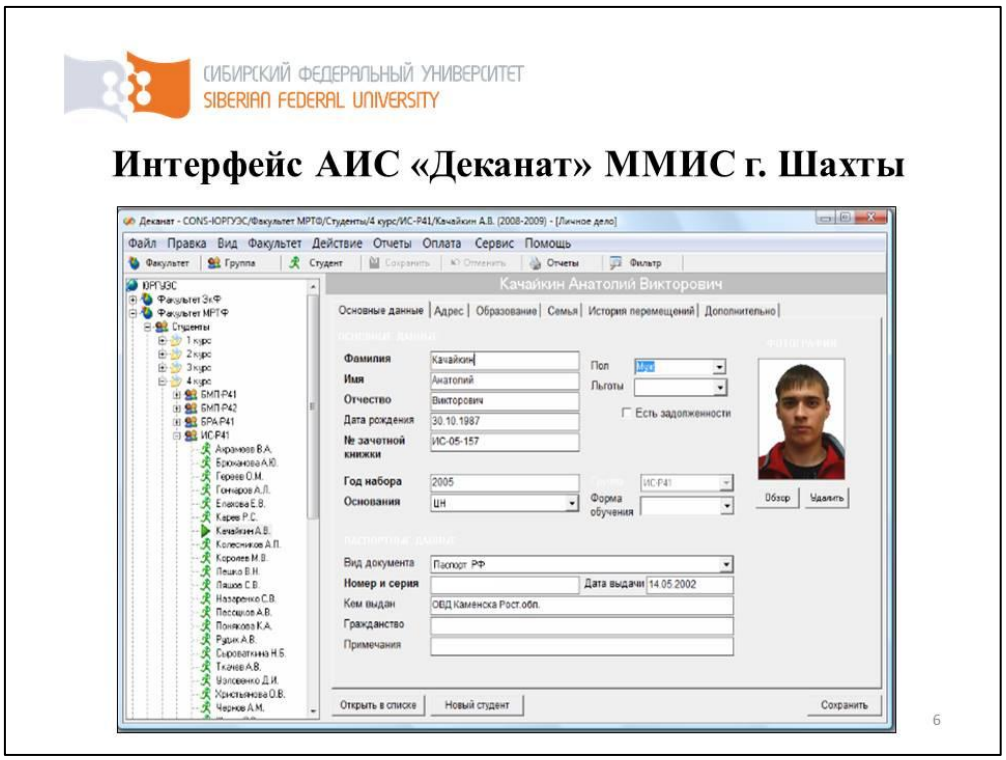


Рисунок В.6 – Слайд презентации № 6

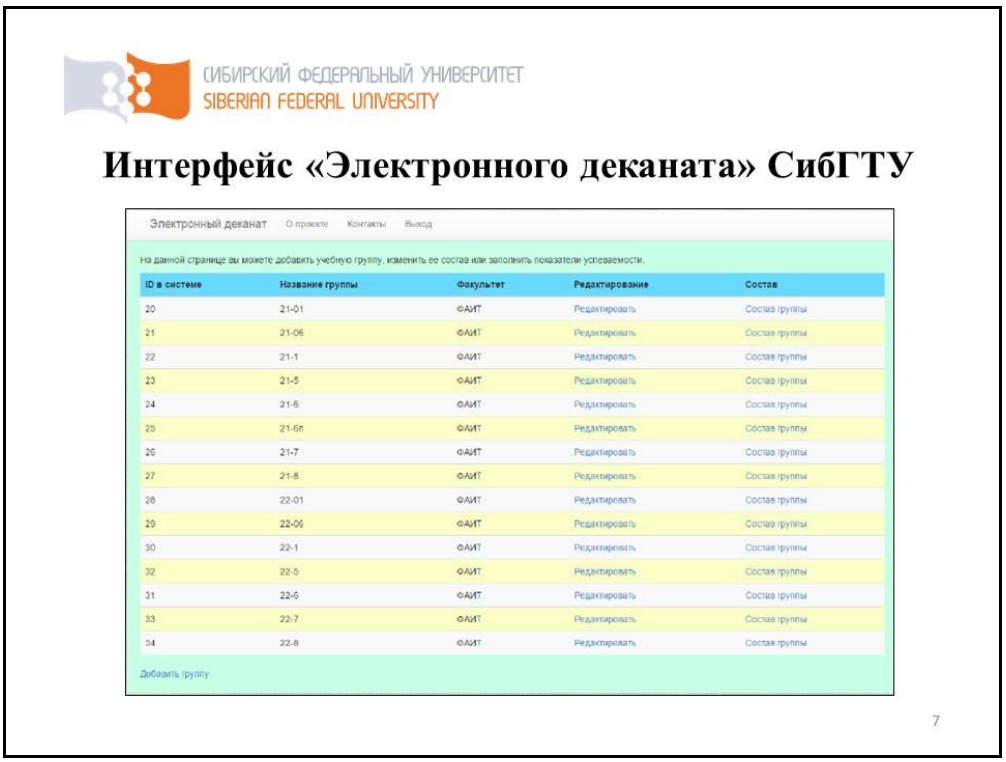


Рисунок В.7 – Слайд презентации № 7

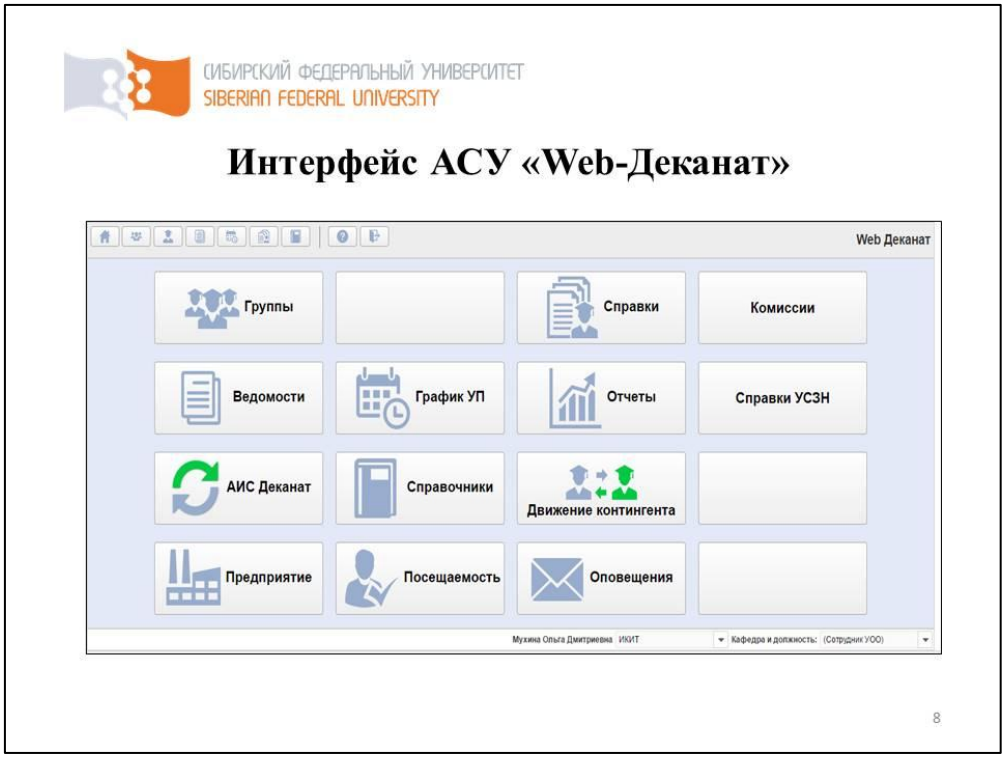


Рисунок В.8 – Слайд презентации № 8

Рисунок В.9 – Слайд презентации № 9

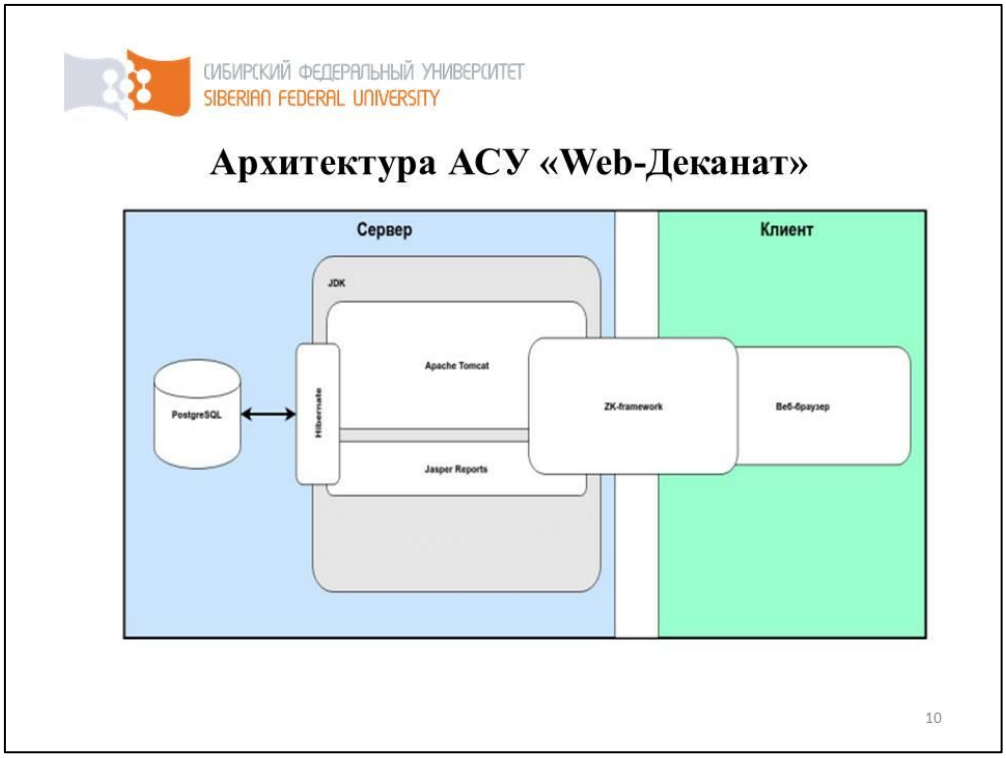


Рисунок В.10 – Слайд презентации № 10

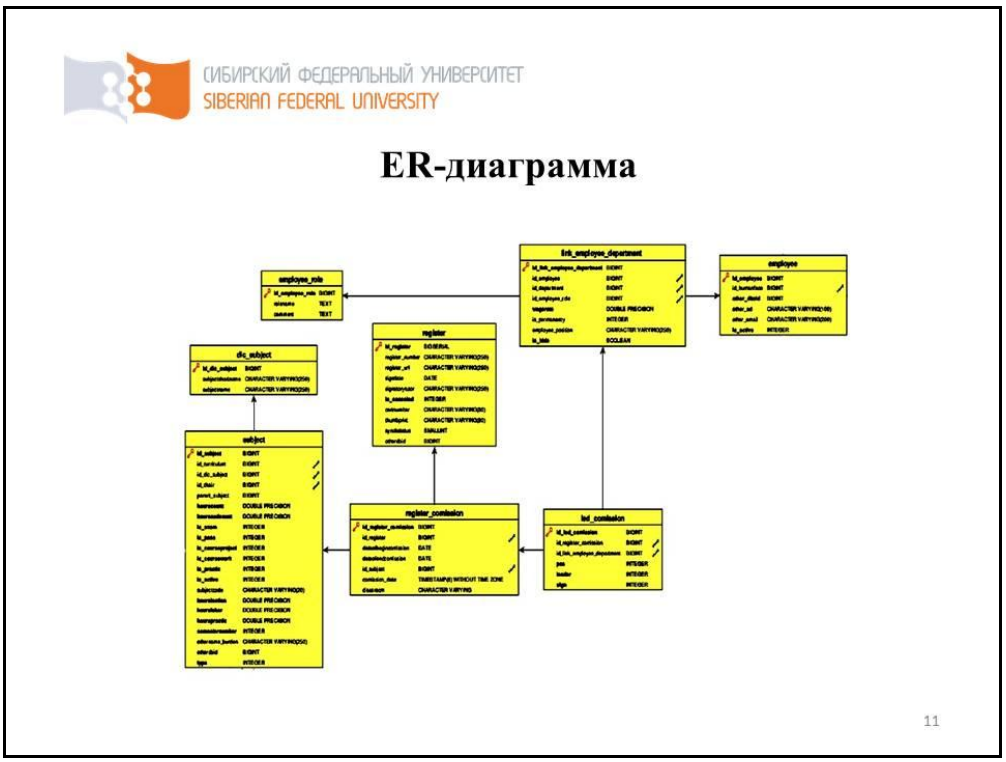


Рисунок В.11 – Слайд презентации № 11

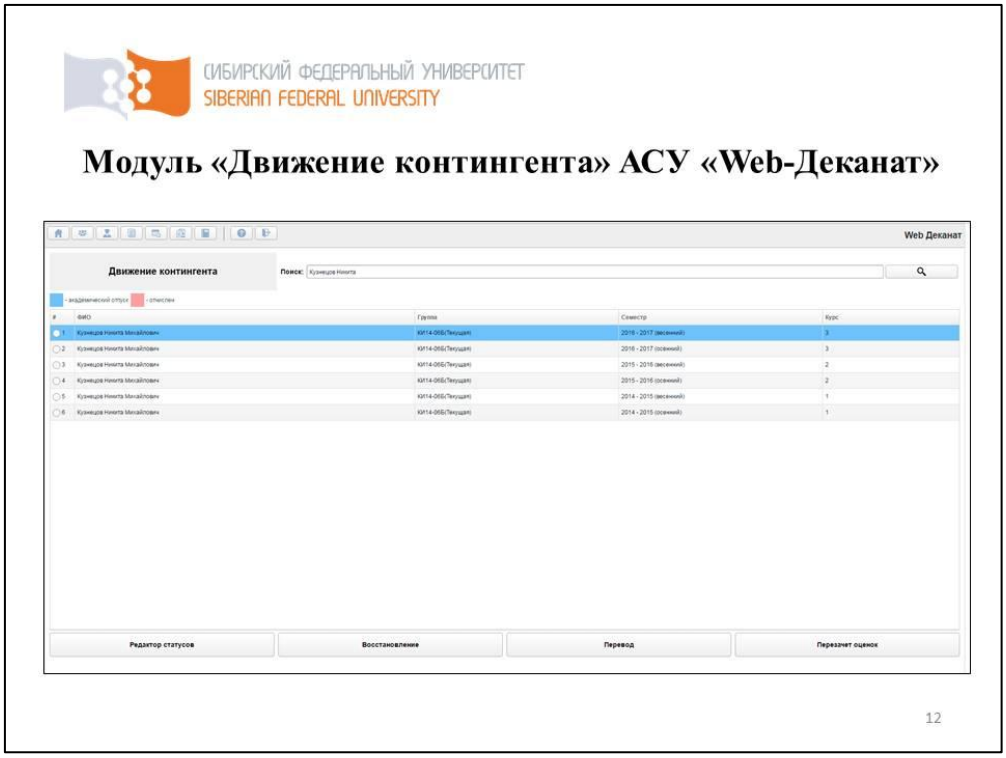


Рисунок В.12 – Слайд презентации № 12

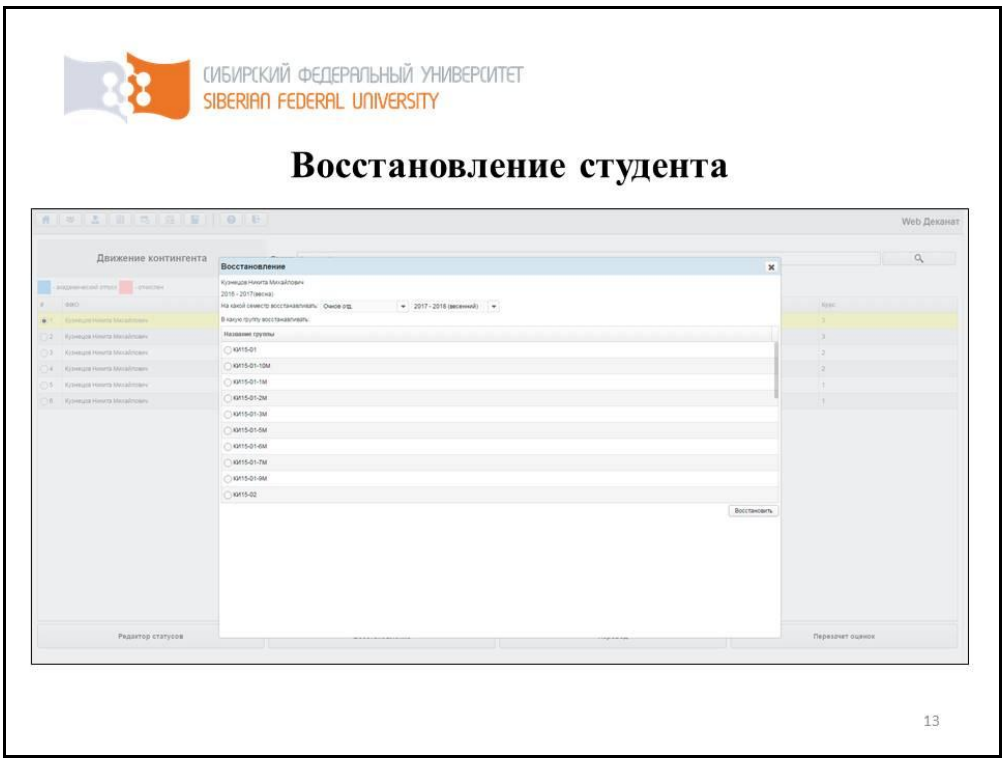


Рисунок В.13 – Слайд презентации № 13

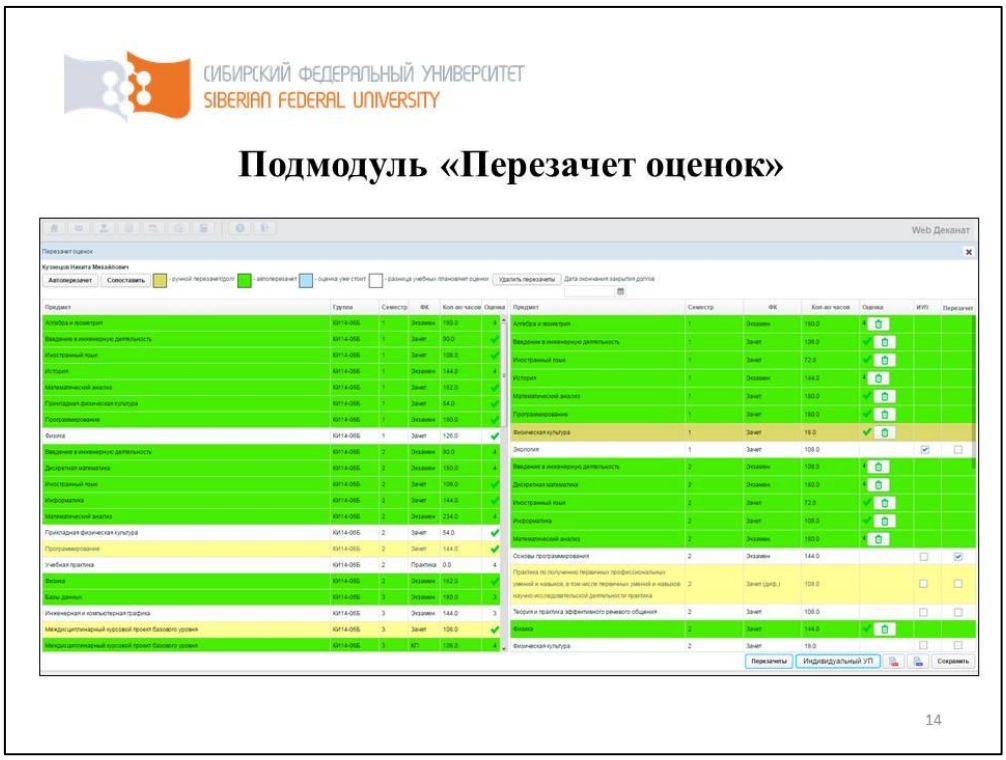


Рисунок В.14 – Слайд презентации № 14

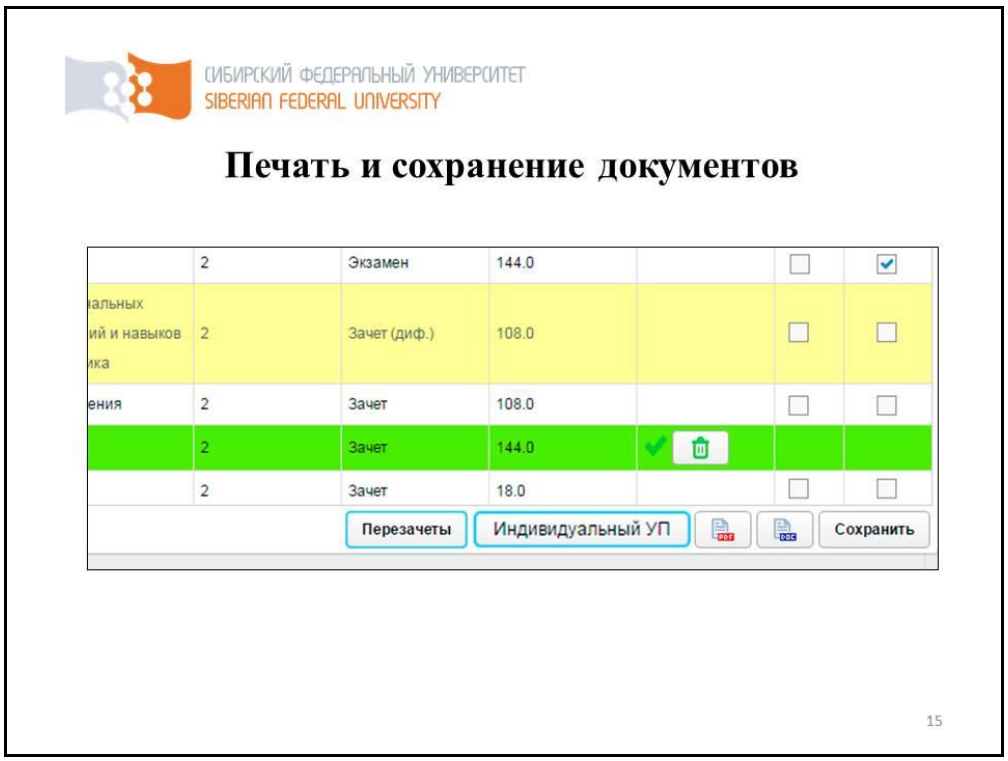


Рисунок В.15 – Слайд презентации № 15

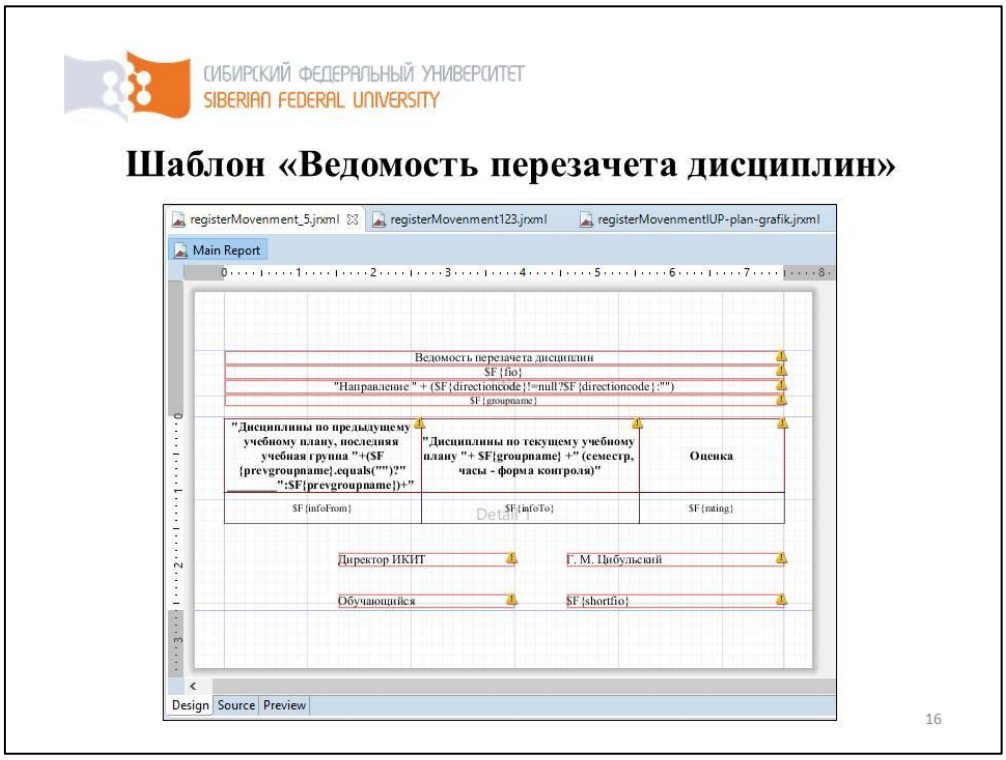


Рисунок В.16 – Слайд презентации № 16

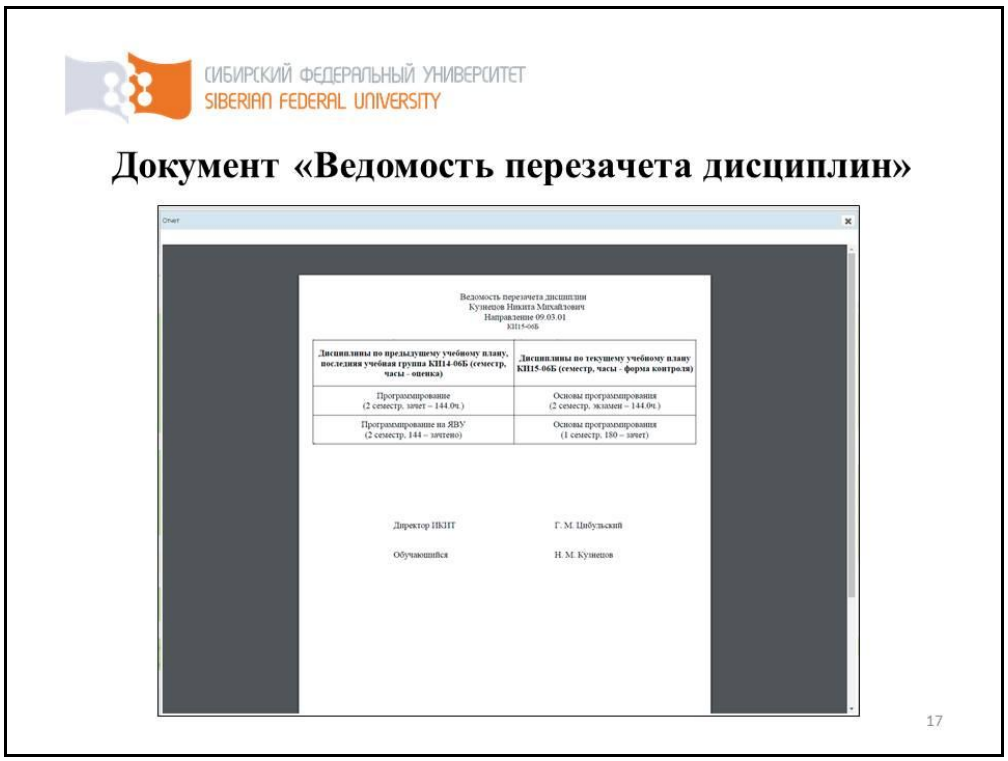
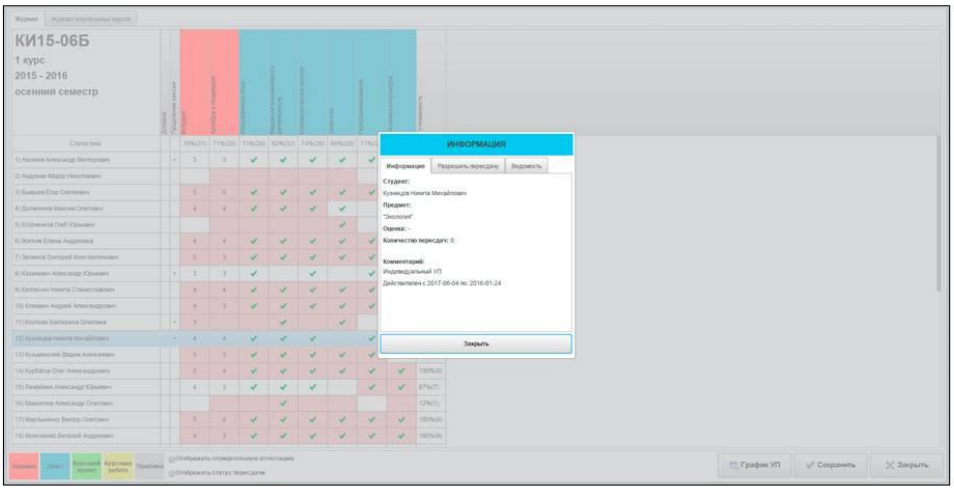


Рисунок В.17 – Слайд презентации № 17

Рисунок В.18 – Слайд презентации № 18

СИБИРСКИЙ ФЕДЕРАЛЬНЫЙ УНИВЕРСИТЕТ
SIBERIAN FEDERAL UNIVERSITY

Информация в паспорте групп о наличие ИУП



19

Рисунок В.19 – Слайд презентации № 19

СИБИРСКИЙ ФЕДЕРАЛЬНЫЙ УНИВЕРСИТЕТ
SIBERIAN FEDERAL UNIVERSITY

Заключение

- проведен анализ предметной области;
- спроектирован подмодуль «Перезачет оценок» в АСУ «Web-Деканат» ИКИТ СФУ;
- разработан подмодуль «Перезачет оценок» в АСУ «Web-Деканат» ИКИТ СФУ;
- стабильная работа системы планируется с 1 сентября 2017 года.

20

Рисунок В.20 – Слайд презентации № 20